175 PTAS
73
mi compuTER
CURSO PRACTICO DEL ORDENADOR PERSONAL,
EL MICRO Y EL MINIORDENADOR
Editorial Delta, S.A.

mi computer CURSO PRACTICO DEL ORDENADOR PERSONAL, EL MICRO Y EL MINIORDENADOR

Publicado por Editorial Delta, S.A., Barcelona

Volumen VII-Fascículo 73

Director:	José Mas Godayol
Director editorial:	Gerardo Romero
Jefe de redacción:	Pablo Parra
Coordinación editorial:	Jaime Mardones
	Francisco Martín
Asesor técnico:	Ramón Cervelló

Redactores y colaboradores: G. Jefferson, R. Ford, F. Martín, S. Tarditti, A. Cuevas, F. Blasco
Para la edición inglesa: R. Pawson (editor), D. Tebbutt (consultant editor), C. Cooper (executive editor), D. Whelan (art editor), Bunch Partworks Ltd. (proyecto y realización)

Realización gráfica: Luis F. Balaguer

Redacción y administración:
Paseo de Gracia, 88, 5.°, 08008 Barcelona
Tels. (93) 215 10 32 / (93) 215 10 50 - Télex 97848 EDLTE

MI COMPUTER, *Curso práctico del ordenador personal, el micro y el miniordenador*, se publica en forma de 96 fascículos de aparición semanal, encuadernables en ocho volúmenes. Cada fascículo consta de 20 páginas interiores y sus correspondientes cubiertas. Con el fascículo que completa cada uno de los volúmenes, se ponen a la venta las tapas para su encuadernación.
El editor se reserva el derecho de modificar el precio de venta del fascículo en el transcurso de la obra, si las circunstancias del mercado así lo exigieran.

ISBN: 84-85822-83-8 (fascículo) 84-7598-067-2 (tomo 7)
84-85822-82-X (obra completa)
Depósito Legal: B. 52-84

Fotocomposición: Tecfa, S.A., Pedro IV, 160, Barcelona-5
Impresión: Cayfosa, Santa Perpètua de Mogoda (Barcelona) 058506

Impreso en España-Printed in Spain-Mayo 1985

Editorial Delta, S.A., garantiza la publicación de todos los fascículos que componen esta obra.

Distribuye para España: Marco Ibérica, Distribución de Ediciones, S.A., Carretera de Irún, km 13,350. Variante de Fuencarral, 28034 Madrid.
Distribuye para Colombia: Distribuidoras Unidas, Ltda., Transversal 93, n.° 52-03, Bogotá D.E.
Distribuye para México: Distribuidora Intermex, S.A., Lucio blanco, n.° 435, Col. San Juan Tlihuaca, Azcapotzalco, 02400, México D.F.
Distribuye para Venezuela: Distribuidora Continental, S.A., Edificio Bloque Dearmas, final Avda. San Martín con final Avda. La Paz, Caracas 1010.

Pida a su proveedor habitual que le reserve un ejemplar de MI COMPUTER. Comprando su fascículo todas las semanas y en el mismo quiosco o librería, Vd. conseguirá un servicio más rápido, pues nos permite realizar la distribución a los puntos de venta con la mayor precisión.

Servicio de suscripciones y atrasados (sólo para España)

Las condiciones de suscripción a la obra completa (96 fascículos más las tapas, guardas y transferibles para la confección de los 8 volúmenes) son las siguientes:

a) Un pago único anticipado de 19 425 ptas. o bien 8 pagos trimestrales anticipados y consecutivos de 2 429 ptas. (sin gastos de envío).

b) Los pagos pueden hacerse efectivos mediante ingreso en la cuenta 6.850.277 de la Caja Postal de Ahorros y remitiendo a continuación el resguardo o su fotocopia a Editorial Delta, S.A. (Paseo de Gracia, 88, 5.°, 08008 Barcelona), o también con talón bancario remitido a la misma dirección.

c) Se realizará un envío cada 12 semanas, compuesto de 12 fascículos y las tapas para encuadernarlos.

Los fascículos atrasados pueden adquirirse en el quiosco o librería habitual. También pueden recibirse por correo, con incremento del coste de envío, haciendo llegar su importe a Editorial Delta, S.A., en la forma establecida en el apartado b).

Para cualquier aclaración, telefonear al (93) 215 75 21.

No se efectúan envíos contra reembolso.





Notas clave

El Music Maker, de Commodore, es un paquete ingeniosamente simple basado en un teclado de piano que se engancha encima del ordenador

Ciertas facilidades de los ordenadores personales invitan al desarrollo de periféricos que las amplíen para aprovechar al máximo su potencial. Esto es especialmente cierto en el caso de las capacidades para gráficos y sonido, para las cuales se ha creado una enorme gama de productos accesorios. Tales periféricos son particularmente necesarios en el Commodore, que carece incluso de las sencillas instrucciones de sonido de que disponen la mayoría de los otros micros. Para esta máquina se han puesto a la venta numerosos paquetes para componer música, pero, a pesar de sus méritos, las melodías aún se deben ejecutar en el teclado tipo máquina de escribir y no en el teclado de piano con el cual están familiarizados los músicos. Sin embargo, con la introducción del Music Maker, Commodore ha producido un paquete que proporciona acceso directo al sumamente veloz chip SID *(Sound Interface Device)* del micro y una cubierta de teclado que permite que el usuario utilice el familiar teclado blanco y negro del piano.

La cubierta, construida en plástico sólido, se encaja con comodidad sobre el cuerpo del teclado del ordenador, más o menos sobre las teclas de función y de máquina de escribir. El teclado Music Maker, que abarca dos octavas, es de un plástico más blando y cada tecla está fijada a la cubierta de forma individual. En la cara inferior de las teclas hay unas proyecciones en forma de dientes que hacen que al pulsar la tecla se haga presión sobre una tecla de máquina de escribir específica de las de abajo, que ha sido programada para producir una cierta nota.

Tratándose de un método tan económico y simple, el sistema funciona sorprendentemente bien. El plástico sólido de la cubierta mantiene con firmeza en su sitio las teclas del piano, y aun cuando el teclado se toque con rapidez de un extremo a otro, es muy raro que se pierdan notas.

El software que viene con el paquete se suministra en cassette o en disco. El programa es activado por menú, principalmente mediante las ocho teclas de función. Éstas permiten al usuario alterar el sonido (o número de sonidos) producido con el teclado de piano, cambiando ya sea la forma de la onda de sonido o bien el tono. Asimismo se proporcionan facilidades para utilizar un ritmo de percusión o bajo a modo de acompañamiento, y un secuenciador permite programar y reproducir melodías. Los sonidos y las frases programadas se pueden guardar (SAVE) o cargar (LOAD) en disco o cassette.

Creador de melodías

El paquete Melody Maker, para el Commodore 64, se compone de una cubierta de teclado de piano y el software para controlarla. La cubierta, de dos octavas, se instala en la estructura plástica del ordenador, de modo que las teclas de piano hacen contacto con las teclas del micro que están programadas para producir las correspondientes notas musicales. La cubierta deja a la vista las teclas de función, dado que éstas son necesarias para programar el sonido. El paquete produce un sonido de gran calidad

Ian McKinnell

Es posible programar ocho voces distintas para producir un sonido diferente. Utilizando la opción F6 (modificar voz), el usuario puede seleccionar la voz a alterar. Luego aparece en la pantalla una serie de opciones, comenzando por *attack* (ataque), *decay* (decaimiento), *sustain* (sostenimiento) y *release* (liberación), eligiéndose un número comprendido entre 0 y 15 para cada uno de estos parámetros. Se le pregunta luego al programador si se han de implementar filtros. (Éstos suprimen sonidos por arriba o por debajo de frecuencias especificadas.) Si uno responde que sí, aparece otra serie de preguntas para determinar su frecuencia y nivel.

El secuenciador es un importante componente de la música sintetizada que, curiosamente, se subvalora en muchos paquetes de música para ordenador. Programado en dos partes, ejecuta una frase repetida. Primero, el programador entra las notas de la frase a través del teclado de piano; después se graba la longitud de cada nota (y, por tanto, el ritmo de la melodía) utilizando la tecla F5.

Los tres ritmos de percusión que se ofrecen son

MUSIC MAKER

SOFTWARE
El software que se suministra con el paquete viene en formato de disco o de cassette

DOCUMENTACION
Se entrega con dos manuales. La guía para el usuario explica lo que es preciso saber para iniciarse, si bien carece de la clase de información detallada que necesitarían los usuarios más experimentados. Un segundo manual, *Start playing keyboard*, ofrece las partituras de 28 melodías populares

VENTAJAS
Por su precio, es un paquete muy valioso, que permite utilizar el Commodore 64 como un verdadero instrumento

DESVENTAJAS
La gama de opciones del paquete es notablemente restringida, en especial con respecto a los ritmos preprogramados de batería y bajo

limitados pero adecuados. Los ritmos de bajo siguen el mismo patrón que la percusión y el programador dispone de tres opciones que le permiten conectar o desconectar el bajo y variar la altura (*pitch*). Según el ritmo que se esté ejecutando, éste bajará la altura de la línea del bajo en un quinto de octava o bien en una octava completa. La velocidad del ritmo se puede alterar pulsando las teclas del cursor: Down (abajo) retarda el compás y Right (derecha) lo acelera.

No obstante, muchas de las opciones no se pueden utilizar juntas. Aunque el bajo y la percusión se pueden ejecutar de forma simultánea, tocando el usuario una melodía en el teclado de arriba, esto sólo se puede realizar con el teclado en modalidad monofónica. En consecuencia, es imposible tocar acordes con ritmo de fondo.

Ello también ocurre con la opción de secuenciador. Muchos grupos *pop* actuales omiten la sección rítmica convencional de bajo y batería y actúan, en cambio, con el apoyo de un secuenciador. En el Music Maker, sin embargo, no existe una facilidad para hacer eso, ni siquiera para reproducir el secuenciador con la sección de ritmo proporcionada. En la actualidad esta limitación se debe al hardware. Dado que el chip SID posee sólo tres voces, uno no puede esperar un ritmo de bajo y percusión y sonido polifónico al mismo tiempo. Con la sección de ritmo no funcionará siquiera el *glissando* (una opción que desliza el sonido de una nota a otra mientras se pulsa la tecla junto con la barra espaciadora), si bien es probable que este problema sea consecuencia del hecho de que el sonido se genera de forma digital. Mantener el bajo y la batería, además de producir un cambio suficientemente pequeño en la altura para producir una ligadura, parece estar más allá de las capacidades del procesador.

Más difícil de comprender es por qué quienes escribieron el software decidieron no permitir que el propio usuario programara sus ritmos de bajo y percusión en la máquina. Las técnicas serían las mismas que las utilizadas en el secuenciador, y una sección de ritmo programable le hubiera añadido al paquete una gran versatilidad.

Otra dificultad que surge al tocar en "tiempo real" es que los parámetros tales como voz y octava no se pueden cambiar mientras el usuario está en "modalidad reproducción". Ello significa que, de hecho, el usuario queda limitado a dos octavas.

Con Music Maker se suministran dos manuales. La guía para el usuario es un folleto con instrucciones para cargar y una sucinta explicación sobre cómo se utilizan cada una de las diversas funciones. No es detallado, pero cumple con su cometido de iniciar al principiante. A partir de ese momento las instrucciones activadas por menú son suficientes para permitir que el usuario desarrolle las capacidades completas del paquete. El otro folleto, *Start playing keyboard*, contiene breves instrucciones acerca de cómo manejar teclados y una explicación de notación musical; incluye, además, las partituras de 28 populares canciones.

El Commodore Music Maker es, ciertamente, un valioso intento por hacer un uso cabal de las facilidades para sonido del Commodore 64. Para quien sea nuevo en el campo de la música por ordenador, el paquete es muy fácil de utilizar y, una vez que uno se acostumbra al teclado, es fácil tocar melodías. Los usuarios más avanzados quizá encuentren que el paquete carece de versatilidad y que el teclado está un tanto apiñado como para permitir efectos complicados. No obstante, una inversión que merece la pena para cualquiera que desee investigar las posibilidades de la música por ordenador.

Menú musical

Esta pantalla muestra la visualización de menú para la opción secuenciador del Music Maker

SEQUENCER
(F1) RECORD NOTES (F2) DELETE NOTE
(F3) RECORD TIME (F5) PLAYBACK
(F7) STOP
(F8) EXIT TO MENU
RECORDING STEP NUMBER 12
A F B G D A G A B B B A
C D E F G A B C D E F G A B

El menú muestra las seis opciones disponibles. Observe que las notas a tocar y su duración (TIME) se entran por separado

Recuerda al usuario que el Music Maker está en modalidad de grabación y no en la de reproducción

Esta línea le recuerda al usuario los nombres de las notas del teclado (C: *do*; D: *re*; E: *mi*; F: *fa*; G: *sol*; A: *la*; B: *si*)

Un cursor azul indica en el teclado qué nota se está tocando en cada momento

Éstas son las notas que se han entrado hasta ahora

El número de paso indica cuántas notas se han entrado hasta el momento en la memoria del secuenciador

Sistema público

Vamos a perfilar los tres sistemas de acceso público más importantes de Gran Bretaña: Telecom Gold, Prestel y Compunet

Los sistemas de acceso público se dividen en dos tipos: los de acceso público gratuitos, por cuya utilización no se le cobra al público ningún cargo, y los sistemas comerciales. Los tres sistemas comerciales más importantes de Gran Bretaña son Telecom Gold, Prestel (que incorpora el Micronet 800) y Compunet. Los sistemas de acceso público gratuitos se conocen comúnmente como tablones de anuncios. Éstos los ejecutan aficionados, en micros normales, para beneficio de otros aficionados. Aquí nos centraremos en los sistemas comerciales y en el próximo capítulo examinaremos con detalle los sistemas de acceso público gratuitos.

Telecom Gold es el propio sistema de correo electrónico de la British Telecom. Dirigido fundamentalmente a los usuarios de gestión, ofrece correo electrónico instantáneo entre usuarios Gold, acceso a facilidades de télex y almacenamiento de archivos. Comprar un "buzón" cuesta 100 libras esterlinas y al cabo del primer mes (que es gratuito) al suscriptor se le cobra tanto por tiempo de permanencia en el sistema como por unidades de almacenamiento. La tarifa mínima es de 10 libras mensuales. La tasa por hora es elevada durante las horas de oficina, disminuyendo su precio alrededor de un tercio en cualquier otro momento. Las tarifas de almacenamiento se hacen por correo sin leer y tanto el correo como los archivos de trabajo son guardados en disco.

Dado que el Gold está diseñado pensando en las grandes empresas, las "contraseñas" de los usuarios se componen de un código de tres letras que identifica a la empresa (conocida como la "familia") y un código de tres dígitos que identifica a la persona dentro de la empresa. Así, por ejemplo, TCG035 identifica a Lucy Storer, de Telecom Gold, quien opera uno de los buzones de "ayuda".

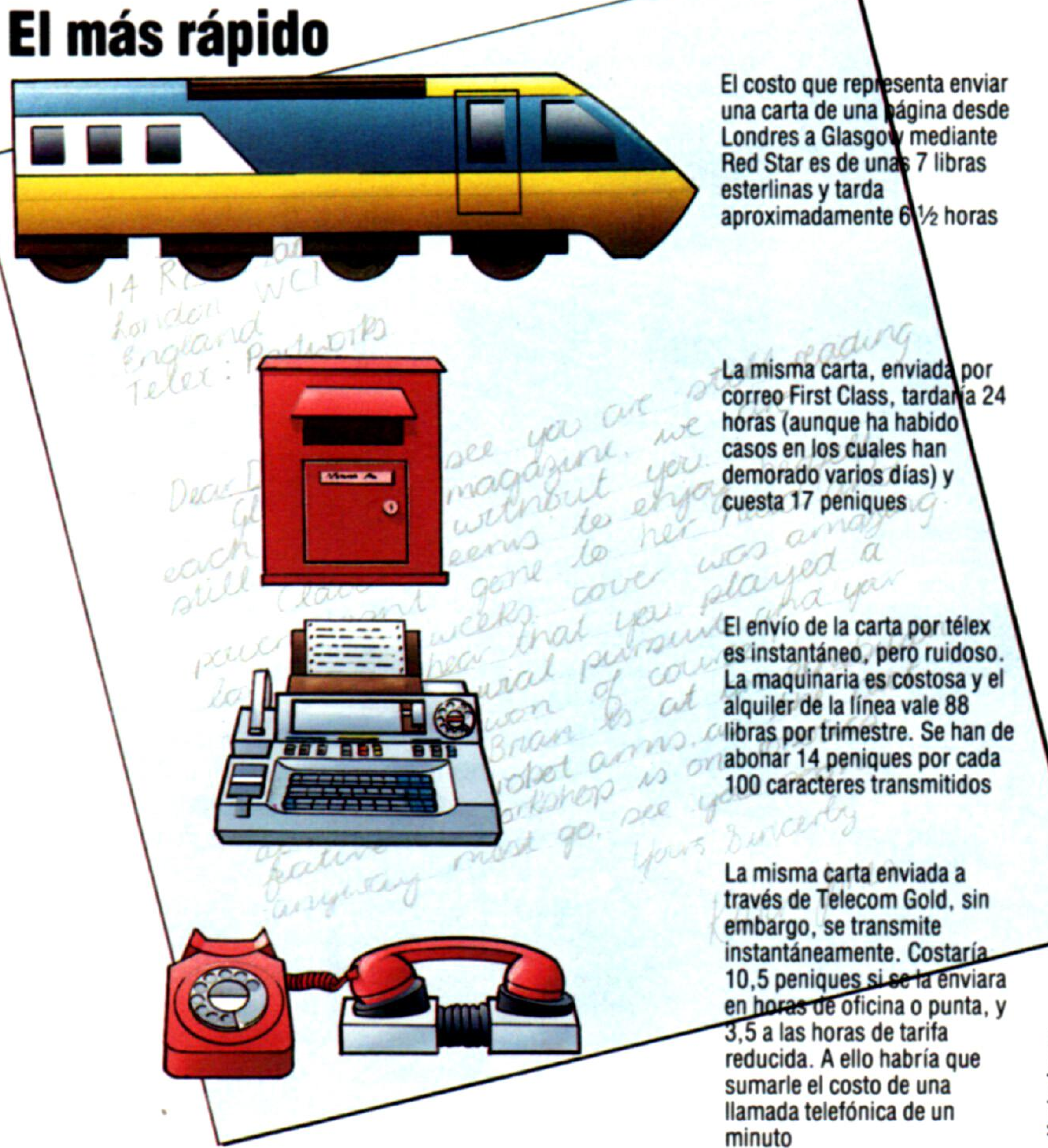
Varias empresas de ordenadores han cogido una gran cantidad de buzones con el sistema de código de tres dígitos y tres letras, a tarifas reducidas, y han vendido luego buzones individuales a sus clientes también a tarifas reducidas. De este modo, los clientes de Tandy pueden adquirir un buzón TCC (Tandy Computer Corporation) por 20 libras en vez de 100. Al adquirirlo de esta manera uno no goza del uso gratuito durante el primer mes; pero, a menos que usted esté seguro de utilizar tarifas por más de 80 libras al mes, conviene averiguar si el fabricante de su micro ofrece una operación similar.

El Prestel también lo lleva British Telecom y, al igual que el Gold, ofrece facilidades de correo electrónico, aunque está diseñado básicamente para proporcionar información. Prestel es una gigantesca base de datos que abarca temas tan diversos como precios de acciones de bolsa, cocina, viajes, listas de éxitos de software, clubs de usuarios de ordenadores, actividades de compañías e informes meteorológicos locales.

El servicio opera según una estructura de árbol: uno empieza en el menú "raíz" y selecciona un área general; informática, por ejemplo. La selección se va estrechando gradualmente a través de una serie de submenús, hasta hallar lo que se estaba buscando: BBC Micro, software, software gratuito, juegos, Pacman. No obstante, si se conoce el número de "página" de la información buscada, pueden saltarse todos los menús e ir a ella directamente.

La principal área de interés para los usuarios aficionados a los ordenadores es la Micronet 800. Ésta permite participar en juegos en línea, cargar software (tanto gratuito como comercial), plantear preguntas técnicas y recibir respuestas, y leer noticias y análisis sobre micros. También puede enviar correo electrónico privado, si bien esta facilidad está relativamente poco perfeccionada.

El más rápido



El costo que representa enviar una carta de una página desde Londres a Glasgow mediante Red Star es de unas 7 libras esterlinas y tarda aproximadamente 6 ½ horas

La misma carta, enviada por correo First Class, tardaría 24 horas (aunque ha habido casos en los cuales han demorado varios días) y cuesta 17 peniques

El envío de la carta por télex es instantáneo, pero ruidoso. La maquinaria es costosa y el alquiler de la línea vale 88 libras por trimestre. Se han de abonar 14 peniques por cada 100 caracteres transmitidos

La misma carta enviada a través de Telecom Gold, sin embargo, se transmite instantáneamente. Costaría 10,5 peniques si se la enviara en horas de oficina o punta, y 3,5 a las horas de tarifa reducida. A ello habría que sumarle el costo de una llamada telefónica de un minuto

Kevin Jones

Hoja de datos de comunicaciones

Sistema	Telecom Gold	Prestel	Compunet	PSS
Número teléfono (datos)	01-278 4355, o a través del PSS	Varios (según la zona) 300/300 baud.=01–248 5747	Varios (según la zona)	Varios (según la zona)
Número teléfono (preguntas)	01-403 6777	01-278 3143	0536-205252	100 (pedir por Freefone PSS)
Costo	£100, más £10 por mes (mínimo)	Alrededor de £100 más £13 trimestrales (mínimo)	£100 más £30 por año	£25 más £25 por trimes (mínimo)
Horas funcionam.	24 horas	24 horas	24 horas	24 horas
Veloc. (baudios)	300/300 y 1200/1200	1200/75 (y 300/300 en Londres)	1200/75	300/300 y 1200/1200
Paridad	Ninguna	Videotexto	Videotexto	Ninguna
Bits de datos	8	Videotexto	Videotexto	8
Bits de final	1	Videotexto	Videotexto	1
Para obtener ayuda	INFO INFO	*0£ (*36£ para dejar mensaje)	Seleccionar HELP del menú	Ninguna disponible
Para desconectarse	OFF	Colgar	Seleccionar LEAVE del menú	Colgar
¿Correo electrón.?	Sí	Sí	Sí	No
¿Mensajes públicos?	Sí	Sí	Sí	No
¿Software comercial?	No	Sí	Sí	No
¿Software gratuito?	No	Sí	Sí	No

Para unirse al Micronet se necesita un modem adecuado y software para comunicaciones, y los usuarios del BBC Micro, el Spectrum o el Commodore 64 pueden obtenerlos de Micronet 800.

Compunet ofrece una gama de servicios similar a la que proporciona el área Micronet 800 del Prestel. El usuario puede enviar correo electrónico, dejar mensajes al público, cargar software y acceder a noticias e información relativa a micros.

Para acceder a Compunet se necesita un modem Commodore. Éste contiene una ROM que contiene el software de comunicaciones necesario para conectar con el sistema, y un número de serie exclusivo. Cuando el suscriptor establece conexión se le solicita su identificación, que se compara con el número de serie de la ROM de su modem: si no coinciden, Compunet no le permitirá el acceso.

Este sistema posee a la vez ventajas e inconvenientes. La ventaja obvia tanto para Compunet como para el usuario es que no importa que alguien le "robe" su código de identificación, porque no podrá utilizarlo sin su modem. La desventaja es que el usuario no puede conectar con el sistema a través del modem de alguna otra persona.

El sistema incorpora otro dispositivo de protección pensado para impedir la piratería de software. Cuando el suscriptor carga software desde Compunet, en el mismo se "estampa" el número de serie del modem utilizado para cargarlo. El software sólo se podrá ejecutar con el modem enchufado.

Al igual que el Prestel, el Compunet se basa en una estructura arborescente, con menús principales que llevan a submenús. El Compunet es, sin embargo, más "amable" que el Prestel, porque ofrece menús en el área principal de la pantalla y un menú de instrucciones que se va desplazando en la parte inferior. El menú de instrucciones ofrece opciones tales como *"VIEW highlighted option"* (ver opción resaltada), *"Go back to the LAST page"* (retroceder hasta la última página), *"COPY the current page to tape or disk"* (copiar la página en curso a cinta o disco), etc. Se utilizan las teclas del cursor arriba/abajo para realizar la selección del menú Compunet, y las teclas izquierda/derecha para hacer la selección del menú de instrucciones.

El Commodore Communications Modem cuesta 100 libras (incluyendo el software) y la tarifa de suscripción a Compunet es de 30 libras al año, siendo gratuitos los primeros 12 meses. El acceso cuesta 7 libras por hora en horario de oficina, siendo gratuito el resto del tiempo (tardes, noches y fines de semana).


El Compunet no es (contrariamente a lo que suele pensar la gente y según la impresión que da la propia Commodore) el servicio de videotexto propio de Commodore. Lo lleva Compunet Teleservices Ltd., una empresa independiente. En la actualidad el sistema sólo soporta al Commodore 64, pero la intención de Compunet es ir soportando de forma gradual todos los micros populares.

Packet Switching Service

Por último, examinemos brevemente el Packet Switching Service (PSS), de la British Telecom, que no es un servicio por ordenador en el sentido del Gold, el Prestel o Compunet, sino una forma económica de acceder a otros sistemas de ordenador.

A menos que usted se encuentre en la zona de llamadas locales del servicio de ordenador que desea utilizar, el principal gasto puede ser el costo de la llamada telefónica. Esto es particularmente cierto si utiliza sistemas norteamericanos, pero también se aplica a las llamadas nacionales de larga distancia. El PSS es la respuesta de la British Telecom a este problema.

El PSS es una red conectada a los principales sistemas de ordenador de acceso público de todo el mundo. Los usuarios pueden conectarse a un "nodo" (punto de entrada) PSS local y acceder, a través del mismo, a cualquiera de los sistemas conectados. El usuario sólo paga el costo de la llamada al nodo local (normalmente, una llamada local) más una tasa PSS. El recuadro "Sesión tipo utilizando el Telecom Gold" muestra cómo se accede al Gold a través del PSS.


Cuando obtenga el tono de portadora, pulse <CR> <CR> a2 <CR>. Esto indica al PSS que Ud. es un terminal CRT, ASCII estándar. PSS responderá entonces con un trozo de código, que será algo parecido a esto:
SLO\A01-7531740139
ntlgold <+non-printing password>
No se le invitará a conectarse; ¡ni siquiera se le ofrecerá un carácter de aviso! El PSS simplemente aguardará a que entre su identificación de usuario de la red (NUI). Ésta es "n" seguido de un ID y una contraseña, ambos de 6 caracteres. Ud. entra todo en una sola línea, que no se visualizará:
ADD?
Si el PSS no reconoce su ID y su contraseña, le responderá con un "NUI?" y esperará que vuelva a probar. Cuando esté conectado, le pedirá la dirección de red del servicio que desea mediante el mensaje:
Luego digite la dirección del servicio que Ud. desea. Ésta será una "a" seguida por un código de 11 dígitos. En este caso, queremos Tel. Gold, su código es
a21920100481
23421920100481+COM
El PSS intenta conectarnos con este servicio. Si todo va bien, visualizará algo así como:
Primecom Network 18.4G System 81
Y Ud. ya está en el sistema pedido. Hemos llamado al Telecom Gold, de modo que veamos cómo funciona:
Please Sign On
>id tcc007
Para conectarse (SIGN ON) hemos de entrar las letras "id" seguidas por una identidad de seis caracteres. Este ID tendrá tres letras describiendo la "familia" de buzones (ver texto principal) y tres números identificando el buzón individual:
Password <non-printing password>
Gold le pedirá entonces la contraseña. Ésta está compuesta por entre 1 y 8 caracteres alfanuméricos, según el caso, y no se visualiza:
TELECOM GOLD Automated Office Services 18.4G(81)
On At 18:23 30/09/84 BST
Last On At 16:53 27/09/84 BST
Gold lo conecta en:
Mail call (1 Unread)
El Gold le dice si en su buzón hay alguna carta y si Ud. ya la ha leído:
>mail
Send, Read or Scan: read
Para leer su correspondencia, entre:
Gold le dice de quién es la carta, de qué se trata, y la fecha y hora en que se envió. Después le pregunta si desea leerla, presentándole un mensaje "MORE?". Pulse <CR> para continuar, "n" para pasar por alto:
To: TCC007 (81:TCC007)
From: L.STORER (BTG035) Posted: Fri 28-Sep-84
16:46 BST UK Sys 80
Subject: Reply to: query re: 81 to 84 changeover
—MORE— <CR>
YES, WHEN I HAVE ALL THE DETAILS I WILL CERTAINLY REPLY
LUCY
Se le preguntará qué desea hacer con la carta. Ud. puede guardarla (SAVE) en sus archivos personales, contestarla (REPLY), eliminarla (DELETE) o dejarla donde está. Se le cargará una tarifa de almacenamiento por las cartas sin leer. Si desea un almacenamiento gratuito, ¡escríbase una carta a Ud. mismo y después léala!
Action Required: reply
Text:
I see. The Helpline tells me that Gold will be doing a mail out to all users informing them of the change. Can you tell me when this will be happening? Will everyone on all systems be notified? If so, it shouldn't be necessary for me to have to tell them personally.
Surya
Ahora enviemos la carta:
send
BTG035 -- Sent
Action Required: delete
End of Mail
Ahora el Gold nos pregunta qué más deseamos hacer con la carta. Dado que ya hemos acabado, la vamos a borrar.
Pulsando <CR> se ahorra la mayoría de los mensajes no deseados.
Send, Read or Scan: <CR>
>mailck
No mail at this time
Si Ud. ha permanecido en el sistema durante un rato, es una buena idea ver si mientras estaba conectado ha llegado alguna otra carta:
Hemos acabado; desconectémonos:
> off
Off At 18:29 30/09/84 BST
Connect Mins = 7
Compute Secs = 4/3
Ahora estamos otra vez en el PSS, pero ante este mensaje Ud. puede simplemente colgar.
CLR DTE (00) 00:00:06:07 47 49
Ian McKinnell



Hundirse o nadar

Nos corresponde analizar el diseño de los dos últimos escenarios especiales de "El bosque encantado": el pantano y el poblado



Aún hemos de considerar dos de los escenarios especiales de *El bosque encantado*: el pantano y el poblado. Comencemos por el pantano. Al igual que para todos los escenarios especiales, es importante decidir sobre una línea argumental antes de ponernos a programar. Esta historia puede ser tan compleja o tan simple como le parezca conveniente al programador, pero ha de tener presente que una trama muy densa para cada escenario puede suponer un enorme esfuerzo de programación y devorar grandes agujeros de la memoria residual del ordenador.



El pantano

La trama argumental para el pantano es la siguiente:

Cuando el jugador entre en el pantano empieza a hundirse. Puede utilizar todas las "instrucciones normales" disponibles, pero no puede salir de la ciénaga.
En cambio, el jugador debe optar por nadar en caso de que quiera salir del pantano.
El jugador sólo puede nadar si lleva consigo menos de dos objetos, puesto que éstos aumentan su peso, contribuyendo a hundirlo.
Si el jugador transporta consigo dos objetos, ha de abandonar uno si no quiere hundirse.
Todos los objetos que se abandonen en el pantano se pierden para siempre.

```
4870 REM **** S/R PANTANO ****
4875 SF=1
4880 SN$="COMIENZAS A HUNDIRTE EN EL PANTANO.":GOSUB 5500
4885 PRINT:INPUT"INSTRUCCIONES";IS$
4890 GOSUB2500:REM DESCOMPONER INSTRUCCION
4895 IF F=0 THEN 4885:REM NO VALIDA
4900 GOSUB3000:REM INSTRUCCIONES NORMALES
4910 IF VB$="MIRAR" THENGOSUB2000:GOTO4885
4915 IF VB$="DEJAR" THEN IV$(F,2)="-2":REM OBJETO PERDIDO
     PARA SIEMPRE
4917 IF VF=1 THEN 4885:REM INSTRUCCION NORMAL
4920 REM ** INSTRUCCIONES NUEVAS **
4925 IF VB$<>"NADAR"THEN SN$="NO COMPRENDO":GOSUB5500:
     GOTO4885
4930 REM ** NADAR **
4932 F=0
4935 FOR I=1 TO 2
4940 IF IC$(I)<>"" THEN F=F+1
4950 NEXT I
4955 IF F<2 THENGOSUB5035:RETURN:REM ALEJARSE NADANDO
4960 GOSUB 5000:RETURN:REM LLEVANDO DOS OBJETOS
5000 REM **** S/R LLEVANDO DOS OBJETOS ****
5010 SN$="LOS OBJETOS SE EMPUJAN HACIA ABAJO Y TE VAS
     HUNDIENDO.":GOSUB5500
5012 PRINT:INPUT"INSTRUCCIONES";IS$
5015 GOSUB2500:REM DESCOMPONER INSTRUCCION
5020 IF VB$<>"DEJAR" THENGOSUB5080:REM HUNDIR
5025 GOSUB 3900:IV$(F,2)="-2":REM ABANDONAR OBJETO
5030 IF HF=0 OR F=0 THEN 5080:SEM HUNDIR
5035 REM **** ALEJARSE NADANDO ****
5040 SN$="AHORA PUEDES CRUZAR EL PANTANO A NADO. EN QUE
     DIRECCION IRAS?":GOSUB5500
5050 SL$(2)="00080605":GOSUB2300:REM DEFINIR Y VISUALIZAR
     SALIDAS
5055 PRINT:INPUT"INSTRUCCIONES";IS$
5060 GOSUB2500:REM DESCOMPONER INSTRUCCION
5062 IF F=0 THEN 5055:REM NO VALIDA
5065 GOSUB3500:REM MOVER
5067 SL$(2)="00000000":REM DATOS SALIDA CERO
5070 RETURN
5075 :
5080 REM **** S/R HUNDIR ****
5085 SN$="TE HUNDES EN EL PANTANO Y TE AHOGAS":GOSUB5500
5090 END
```



Las líneas 4870-4917 permiten manejar las instrucciones normales, empleando las subrutinas estándares diseñadas con anterioridad. Si el jugador elige dejar un objeto inmediatamente al entrar en el pantano, se ocupará de ello la rutina DEJAR. Sin embargo, esta rutina reintegra la posición del objeto dejado al inventario principal, IV$(,), utilizando el valor en curso del contador de escenarios, P. Ello a su vez significa que, por cuanto atañe al programa, el objeto dejado se halla ahora en el escenario del pantano. Si deseamos perder todo rastro de un objeto que sea abandonado en la ciénaga, en el caso de que un jugador desee dejar aquí un objeto, hemos de corregir la entrada correspondiente en IV$(,).

Recuerde que IV$(F,2) contiene la posición del objeto F. Normalmente éste es el número de escenario, o −1 si el objeto lo lleva consigo el jugador. Para lograr que el objeto desaparezca por completo del juego, hemos de hacer que IV$(F,2) no pueda ser interpretada como un escenario ni indicar que el jugador tiene el objeto. En la línea 4915, IV$(F,2) toma el valor −2, habiéndosele ya asignado a F el objeto en cuestión mediante la rutina DEJAR.

Si el jugador opta por NADAR, el programa baja hasta la línea 4930. Aquí se comprueba el inventario personal del jugador para determinar cuántos objetos lleva consigo. Si el contador es menor que dos, entonces el programa llama a la subrutina ALEJARSE NADANDO, que permite que el jugador salga del pantano. Si el jugador está transportando dos objetos, se le ofrece la oportunidad de deshacerse de uno; si no sigue este curso de acción, el jugador se hunde.

La rutina ALEJARSE NADANDO permite al jugador especificar en qué dirección quiere nadar. Los datos de salida que se dan para el pantano en las sentencias DATA originales son 00000000, que indican que la ciénaga no tiene salida. La línea 5050 vuelve a definir los datos de salida y llama a la subrutina que las describe. Al jugador se le permite entonces seleccionar una y, por consiguiente, huir del pantano. Observe que al final de esta rutina los datos de salida para el pantano se ponen a cero, de modo que si posteriormente el jugador volviera a entrar en la ciénaga, no habría salida.

El poblado

El poblado ofrece el camino para escapar del bosque encantado, si bien el jugador todavía debe lle-

var a cabo algunas tareas antes de que se le permita abandonar la aventura.

La línea argumental para el poblado es ésta:

El poblado está rodeado por una muralla alta y aparentemente imposible de escalar.
La puerta de la muralla está vigilada.
Para entrar en el poblado el jugador debe primero matar al guardián, utilizando la escopeta.
El jugador debe entonces abrir la cerradura de la puerta mediante la llave, para huir del bosque.

La rutina del poblado consta de dos partes: primero, se ha de sortear el peligro del guardián y, segundo, se ha de abrir la puerta. No es difícil imaginar un escenario en el cual el jugador, llevando la escopeta, llegue al poblado y mate al guardián, sólo para descubrir que se necesita una llave para abrir la puerta. Si el jugador no posee la llave, entonces habrá de abandonar el escenario del poblado en busca de ella. Si, al regresar el jugador, se ofrece la misma descripción del escenario del poblado (es decir, que hay un guardián custodiando la puerta), ello no será coherente. Se debe afrontar y resolver un problema particular del juego una sola vez. Si el guardián es muerto, entonces se debe suprimir esa "característica" del poblado.

Esto no es tan difícil como parece. Se puede emplear una bandera que señale si el guardián está vivo o muerto. Con este fin se utiliza la bandera GF: su valor inicial es cero, indicando que el guardián está vivo. Si el jugador mata al guardián, entonces GF se pone a 1. El valor de GF se puede comprobar al entrar en el escenario del poblado para determinar si aún existe el peligro del guardián o no. Habiendo muerto el guardián, el jugador avanza hacia la puerta.

Las instrucciones se pueden descomponer y manejar mediante las subrutinas estándares que ya poseemos. Si el jugador posee la llave y la utiliza para abrir la puerta, habrá completado con éxito la aventura.

```
5100 REM **** S/R POBLADO ****
5102 SF=1
5105 SN$="EL POBLADO ESTA RODEADO POR UNA ALTA MURALLA
     .":GOSUB5500
5106 IF GF<>0 THEN GOSUB5190:RETURN:REM PUERTA
5107 SN$="HAY UN GUARDIAN JUNTO A LA PUERTA DE ACCESO AL
     POBLADO":GOSUB5500
5115 PRINT:INPUT"INSTRUCCIONES";IS$
5120 GOSUB2500:IF F=0 THEN 5115:REM NO VALIDA
5125 GOSUB3000:REM INSTRUCCIONES NORMALES
5130 IF VB$="MIRAR" THEN GOSUB2000:REM DESCRIBIR
5135 IF VB$="AVANZAR" AND MF=1 THEN RETURN
5140 IF VF=1 THEN 5115:REM SIGUIENTE INSTRUCCION
5145 IF VB$<>"MATAR" THEN SN$="NO COMPRENDO":GOSUB-
     5500:GOTO5115
5150 REM ** MATAR **
5155 SN$="QUE USARAS PARA MATAR AL GUARDIAN?":
     GOSUB5500
5160 SN$="ENTRE OBJETO 0 <I> PARA INSTRUCCION":
     GOSUB5500
5162 INPUT IS$:IF IS$="I" THEN 5115
5165 GOSUB2500:REM DESCOMPONER
5167 IF F=0 THEN 5160:REM NO VALIDA
5170 GOSUB5300:IF F=0 THEN SN$="NO HAY NINGUNA
     "+W$:GOSUB5500:GOTO5160
5172 OV=F:GOSUB5450:REM LLEVA CONSIGO EL OBJETO
5174 IF HF=0 THEN SN$="TU NO TIENES
     LA "+IV$(F,1):GOSUB5500:GOTO5160
5175 IF F<>1 THEN SN$="LA "+IV$(F,1)+" NO SIRVE":GOSUB-
     5500:GOTO5160
5180 SN$="MATAS AL GUARDIAN":GOSUB5500:GF=1
5185 :
5190 REM **** S/R PUERTA CERRADA CON LLAVE ****
5195 SN$="DEBES AVANZAR Y TRATAR DE ABRIR LA PUERTA DE
     ENTRADA AL POBLADO"
5200 SN$=SN$+" PERO LA PUERTA ESTA CERRADA Y NO SE MUEVE
     ":GOSUB5500
5205 PRINT:INPUT"INSTRUCCIONES":IS$
5210 GOSUB2500:IF F=0 THEN 5205:REM NO VALIDA
5215 GOSUB 3000:REM INSTRUCCIONES NORMALES
5220 IF VB$="MIRAR" THEN GOSUB2000:REM DESCRIBIR
5225 IV VB$="AVANZAR" AND MF=1 THEN RETURN
5230 IF VF=1 THEN 5205:REM SIGUIENTE INSTRUCCION
5232 IF VB$="USAR" THEN 5240
5234 IF VB$="ABRIR"THEN SN$="COMO?":GOSUB5500:
     GOTO5205
5235 SN$="NO COMPRENDO":GOSUB5500:GOTO5205
5240 GOSUB5300:REM OBJETO VALIDO
5242 OV=F:GOSUB5450:REM LLEVA OBJETO CONSIGO
5244 IF F=0 THEN SN$="NO HAY
     NINGUNA "+W$:GOSUB5500:GOTO5205
5246 IF HF=0 THEN SN$=TU TIENES LA
     "IV$(F,1):GOSUB5500:GOTO5205
5248 IF F<>3 THEN SN$="LA "+IV$(F,1)+" NO SIRVE":GOSUB-
     5500:GOTO5205
5250 REM ** ATRAVESAR LA PUERTA Y A SALVO **
5255 SN$="ABRES LA PUERTA Y, DISFRAZANDOTE CON
     LA ROPA"
5260 SN$=SN$+" DEL GUARDIAN MUERTO, ATRAVIESAS EL
     POBLADO SIN SER DESCUBIERTO"
5265 SN$=SN$+" Y LLEGAS A LA SEGURIDAD DEL MUNDO
     EXTERIOR.":GOSUB5500
5270 END
```

Para poder llamar a estas dos rutinas de escenario especial debemos modificar la línea 2720 de la subrutina que decide si un escenario es o no especial. Introduzca esta modificación:

```
2720 ON P GOSUB4590,4870,5100,4590
```

Complementos al BASIC

Spectrum

En el listado de *El bosque encantado*, sustituya SN$ por S$, IS$ por T$, VB$ por B$, IV$(,) por V$(,), SL$() por X$() e IC$() por I$().
Reemplace las siguientes líneas:

```
2720 IF P=1 THEN GOSUB 4590
2722 IF P=2 THEN GOSUB 4870
2724 IF P=3 THEN GOSUB 5100
2726 IF P=4 THEN GOSUB 4590

4937 LET A$=IC$(I):GOSUB7000
4940 IF A$<>"" THEN LET F=F+1

5175 IF F<>1 THEN LET S$="LA":
     LET A$=V$(F,1):GOSUB7000
5177 IF F<>1 THEN LET S$=S$+"NO
     SIRVE":GOSUB5500:GOTO5160

5248 IF F<>3 THEN LET S$="LA"
     :LET A$=V$(F,1):GOSUB7000
5249 IF F<>3 THEN LET S$=S$+"NO
     SIRVE":GOSUB5500:GOTO5205
```

En el listado de *Digitaya*, reemplace SN$ por S$, IS$ por I$, VB$ por B$, IV$(,) por V$(,) e IC$() por I$(). Sustituya estas líneas:

```
3650 LET S$="TU ":LET A$=V$(F,1):
     GOSUB 7000
3655 LET S$=S$+" NO SIRVE DE
     NADA, LA FUERZA AUMENTA"

4520 LET V$(4,2)=STR$(INT(RND(1)
     *40+8)
```

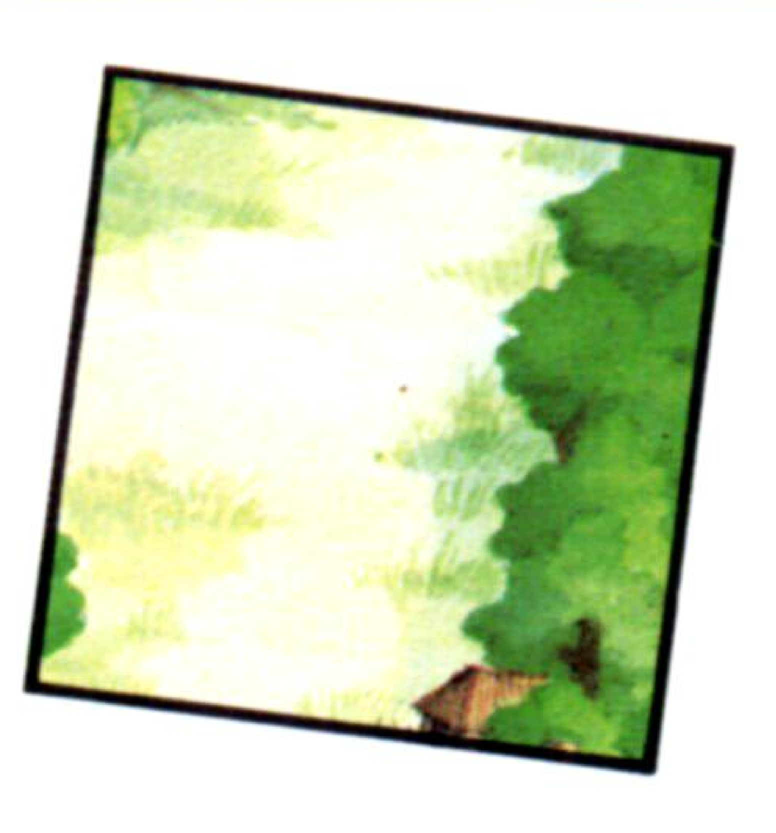
BBC Micro:

En el listado de *El bosque*... sustituya esta línea:

```
190 LET GF=0
```

En el listado de *Digitaya* reemplace esta línea:

```
4520 IV$(4,2)=RND(40)+8
```













Listado de "Digitaya"

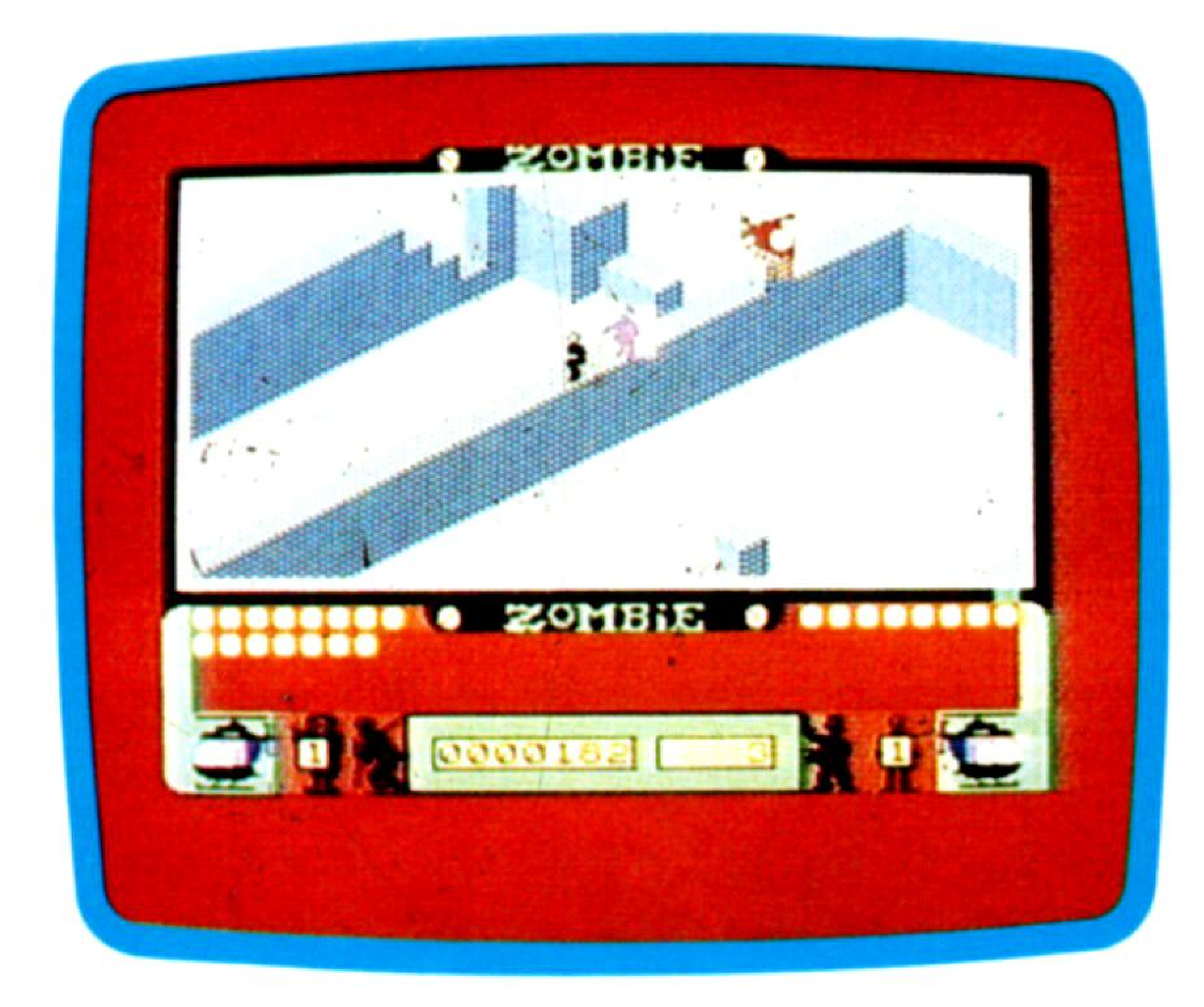
```
2960 REM **** S/R PUERTA PARA EL USUARIO ****
2970 SF=1
2980 SN$="EL ESCAPE ESTA A TU ALCANCE PERO EL VENDEDOR
     DE BILLETES DEL RDD"
2990 SN$=SN$+" TE IMPIDE EL PASO, TE DICE QUE TIENE
     INSTRUCCIONES DE"
3000 SN$=SN$+" ACEPTAR SOLO ENTRADAS. NO OBSTANTE
     ACEPTA LAS PRINCIPALES"
3010 SN$=SN$+" TARJETAS DE CREDITO."
3020 GOSUB 5880:REM FORMATEAR SALIDA
3030 :
3040 PRINT:INPUT"INSTRUCCIONES";IS$
3050 GOSUB1700:REM ANALIZAR INSTRUCCIONES
3060 GOSUB 1900:REM ACCIONES NORMALES
3070 IF MF=1 THEN RETURN:REM IRSE
3080 IF VF=1 THEN 3040:REM SIGUIENTE INSTRUCCION
3090 IF VB$<>"DAR" THENPRINT"NO COMPRENDO":GOTO3040
3100 :
3110 REM ** LA INSTRUCCION ES DAR **
3120 GOSUB5730:REM ES VALIDO EL OBJETO
3130 IF F=0 THEN PRINT"NO HAY NINGUN ";NN$:GOTO3040:
     REM SIGUIENTE INSTRUCCION
3140 :
3150 REM ** ES EL OBJETO TARJETA DE CREDITO **
3160 IF F<>5 THEN PRINT"SOLO ACEPTA TARJETAS DE
     CREDITO":GOTO3040
3170 :
3180 REM ** LLEVA TARJETA CONSIGO **
3190 OV=5:GOSUB5830
3200 IFHF=0THENPRINT"NO TIENES LA ";IV$(5,1):GOTO 3040
3210 :
3220 SN$="EL EMPLEADO COGE LA TARJETA Y DICE 'ESTA
     SERVIRA, SEÑOR'"
3230 GOSUB5880:REM FORMATEAR SALIDA
3240 SN$="SE TE PERMITE PASAR LA BARRERA Y ENTRAR EN
     LA PUERTA PARA EL USUARIO"
3250 GOSUB5880:REM FORMATEAR SALIDA
3260 :
3270 REM ** LLEVA CONSIGO EL DIGITAYA **
3280 OV=6:GOSUB5830
3290 IF HF=1 THEN 3380:REM EXITO
3300 :
3310 REM ** FRACASO **
3320 SN$="BIEN HECHO HAS LOGRADO ESCAPAR DE LAS
     GARRAS"
3330 SN$=SN$+" DE LA MAQUINA, PERO HAS FRACASADO EN
     TU MISION"
3340 SN$=SN$+" CONSISTENTE EN RECUPERAR EL
     MISTERIOSO DIGITAYA"
3350 GOSUB5880:REM FORMATEAR SALIDA
3360 END
3370 :
3380 REM ** EXITO **
3390 SN$="FELICITACIONES, HAS CUMPLIDO CON TU MISION"
3400 SN$=SN$+" DE RESCATAR AL MARAVILLOSO DIGITAYA
3410 SN$=SN$+" DE LAS GARRAS DE LA MAQUINA."
3420 GOSUB5880:REM FORMATEAR SALIDA
3430 END
3440 :
3450 REM **** S/R PUERTA PARA CASSETTE ****
3460 SF=1
3470 SN$="SIENTES UNA FUERZA IRRESTIBLE QUE TE EMPUJA
     HACIA"
3480 SN$=SN$+" UNA SUSPENSION MAGNETICA
     PERMANENTE"
3490 GOSUB5880:REM FORMATO
3500 NS=0:REM COMENZAR A CONTAR LAS INSTRUCCIONES
3510 REM ** INSTRUCCIONES **
3520 NS=NS+1:IFNS>3THEN3770:REM ABSORBIDO
3530 PRINT:INPUT"INSTRUCCIONES";IS$
3540 GOSUB1700:REM ANALIZAR INSTRUCCIONES
3550 GOSUB1900:REM ACCIONES NORMALES
3560 IF MF=1 THEN MF=0:PRINT"NO PUEDES
     MOVERTE...TODAVIA":GOTO3510
3570 IF VF=1 THEN 3510:REM SIGUIENTE INSTRUCCION
3580 IF VB$<>"USAR" THENPRINT"NO COMPRENDO":
     GOTO3510
3590 REM ** LA INSTRUCCION ES USAR **
3600 GOSUB5730:REM ES VALIDO EL OBJETO
3610 IF F=0 THENPRINT"NO HAY NINGUN ";NN$:GOTO3510
3620 :
3630 REM ** ES EL OBJETO EL ACTIVADOR DEL BUFFER **
3640 IF F=8 THEN3680:REM OK
3650 SN$="TU "+IV$(F,1)+" NO SIRVE DE NADA, LA FUERZA
     SE INTENSIFICA"
3660 GOSUB5880:GOTO3510:REM SIGUIENTE INSTRUCCION
3670 :
3680 OV=8:GOSUB5830:REM LLEVA CONSIGO ACTIVADOR
     BUFFER
3690 IF HF=0THEN SN$="NO TIENES EL "+IV$(8,1):
     GOSUB5880:GOTO3510
3700 :
3710 REM ** SALVADO **
3720 SN$="UTILIZAS EL ACTIVADOR DEL BUFFER PARA
     CONTRARRESTAR EL EMPUJE"
3730 SN$=SN$+" HACIA EL OLVIDO MAGNETICO. LA FUERZA
     DISMINUYE"
3740 GOSUB5880:REM FORMATEAR
3750 RETURN
3760 :
3770 REM ** ABSORBIDO **
3780 SN$="LA FUERZA SE VUELVE DEMASIADO INTENSA Y ERES
     EMPUJADO"
3790 SN$=SN$+" A TRAVES DE LA PUERTA PARA CASSETTE A
     LA NADA MAGNETICA."
3800 GOSUB5880:REM FORMATEAR
3810 END
4180 REM **** S/R DISPOSITIVO TRIESTADO ****
4190 SF=1
4200 SN$="UN ENORME CARTEL REZA 'E/S POR AQUI' PERO
     CUANDO TE ESTAS ACERCANDO AL MISMO"
4210 SN$=SN$+" UN REVISOR GRITA 'BILLETES POR FAVOR'"
4220 GOSUB5880:REM FORMATEAR
4230 :
4240 REM ** INSTRUCCIONES **
4250 PRINT:INPUT"INSTRUCCIONES";IS$
4260 GOSUB1700:GOSUB1900:REM ANALIZAR
4270 IFMF=1 THEN RETURN
4280 IFVF=1 THEN4240:REM SIGUIENTE INSTRUCCION
4290 IF VB$<>"DAR"ANDVB$<>"OFRECER"THEN PRINT"NO
     COMPRENDO":GOTO4240
4300 REM ** LA INSTRUCCION ES DAR **
4310 GOSUB5730:REM ES VALIDO EL OBJETO
4320 IFF=0THENPRINT"NO HAY NINGUN ";NN$:GOTO4240:REM
     SIGUIENTE INSTRUCCION
4330 :
4340 REM ** ES UN TICKET EL OBJETO **
4350 IFF=4 THEN4400:REM OK
4360 SN$="EL REVISOR SACUDE LA CABEZA Y DICE"
4370 SN$=SN$+" 'NO PUEDO ACEPTAR ESTE' +IV$(F,1)
4380 GOSUB5880:GOTO4240:REM SIGUIENTE INSTRUCCION
4390 :
4400 OV=4:GOSUB5830:REM LLEVA EL BILLETE CONSIGO
4410 IFHF=0THEN PRINT"NO TIENES EL BILLETE":GOTO4240
4420 :
4430 REM ** OK **
4440 SN$="EL REVISOR ACEPTA TU BILLETE Y TE PERMITE"
4450 SN$=SN$+" TRASPASAR LA BARRERA."
4460 GOSUB5880:REM FORMATEAR
4470 REM ** ELIM. BILLETE DE LA LISTA **
4480 F=0
4490 FORJ=1TO4
4500 IF IC$(J)=IV$(4,1)THENIC$(J)="":J=4
4510 NEXT J
4520 IV$(4,2)=STR$(INT(RND(TI)*40+8)):REM REASIGNAR
     POSICION BILLETE
4530 P=15:MF=1:RETURN
```

Yo, el jurado

Nuestros comentaristas hacen una valoración de dos programas de juegos escritos para el Sinclair Spectrum

Zombie zombie

Quicksilva



Dave Nicholls: ¡Oh, no! Antescher ha sido invadida por los zombies, zombies verdes y horribles que se ponen rojos por la furia y te atacan cuando te acercas demasiado a ellos.

Para ayudarte a librar a la ciudad de esta marea de fanáticos de Duran Duran, dispones de un helicóptero con el cual puedes volar con total seguridad. Pero debes abandonarlo para destruir a los zombies, ¡hazaña que se consigue cogiéndolos por detrás! Cuando les haces blanco, se convierten en una especie de esclavos y los puedes conducir hasta cualquier pared, desde donde saltarán hacia la muerte al son de *Ten green bottles* (Diez botellas verdes). Si no estás muy seguro de poder controlar a los zombies de este modo, puedes dispararles con tu *puffer*, lo que les alejará hasta una distancia segura.

Básicamente se trata de otro *Ant attack* (Ataque de las hormigas) y por ello les gustará a los usuarios que disfrutaron con este juego. Pero si *Ant attack* te resultó una experiencia soporífera, no te molestes en probar el *Zombie zombie*. No obstante, si no has probado ninguno de los dos, probablemente el mejor de ambos sea el que ahora comentamos. *3/5* ÉXITO

Ross Holman: Es demasiado parecido a *Ant attack* como para ofrecer nada emocionante. Las melodías son atrayentes y te lo pasarás bien escribiendo *graffiti* en los ladrillos. *1/5* FIASCO

Roger Willis: Perseguir a los resucitados puede ser divertido, aun cuando sea un entretenimiento de escasa originalidad. *3/5* ÉXITO

Eddie Kidd jump challenge

Martech



Roger Willis: Puedo afirmar, por mi propia experiencia, que inclinarte hacia adelante o hacia atrás cuando estás saltando en motocicleta puede afectar seriamente tu estabilidad. También puedo asegurar que, tal como sucede en este programa, hacerlo mal supone unas consecuencias desastrosas.

La distancia de aproximación parece irrelevante, dado que al despegar de la rampa la motocicleta ha puesto sus cinco marchas y la velocidad indicada es la máxima. En realidad, todo lo que tienes que hacer es controlar la moto en el aire y eso, presumiblemente, determina la cantidad de coches sobre los que consigues saltar con éxito. Suena bien, ¿no? Bueno, pues no es así...

Los mezquinos gráficos sólo sirven para subrayar el hecho de que Eddie Kidd le ha dado su nombre a un juego recreativo bastante primitivo que se vuelve aburrido antes de que el jugador consiga dominarlo. La pantalla de introducción (o de práctica) exige que saltes sobre barriles de gasolina; ¡lo malo es que tus saltos siempre son buenos! Oh, sí, más allá de un cierto límite los saltos producen una invitación a SAVE (guardarlos) en una cinta de cassette virgen con el objeto de participar en competiciones; pero, yo, en realidad, no creo que nadie se moleste en hacerlo. *1/5* FIASCO

Ross Holman: Puedes controlar a Eddie bastante bien; pero incluso después de cierta práctica me resultó difícil realizar saltos con éxito en distancias relevantes. ¡Las "ráfagas de viento" tampoco me resultaron de ayuda! *2/5* FIASCO

Para los más pequeños



Las primeras palabras de un niño
My Talking Computer está pensado como la primera introducción del niño al ordenador. Está diseñado para que su utilización sea lo más sencilla posible. Las cubiertas, incluidas en un libro de anillas, se colocan sobre un teclado sensible al tacto y se sujetan mediante un marco. En la ranura situada en el lado izquierdo se pueden insertar fácilmente módulos de ampliación que contienen programas extras

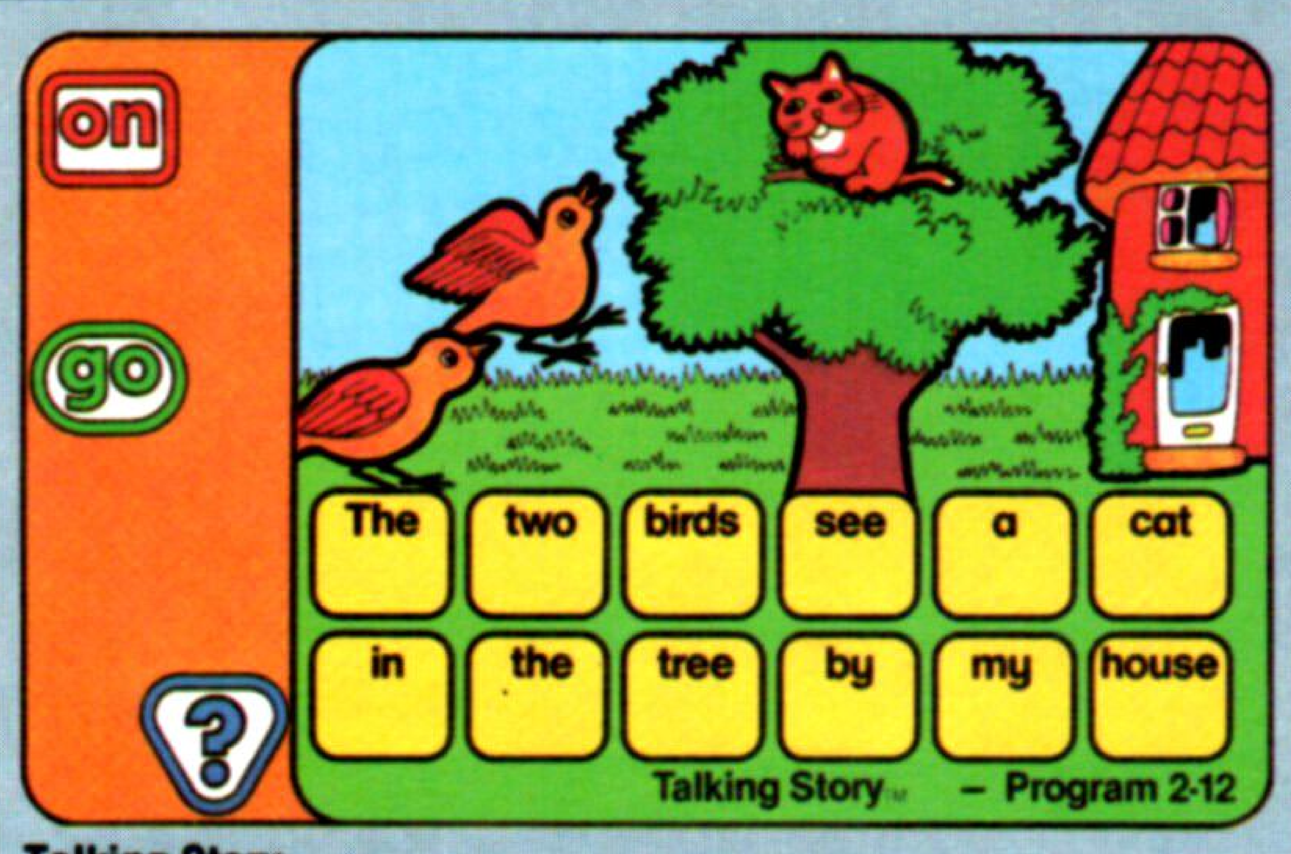
My Talking Computer es un "ordenador" destinado a la enseñanza de niños en edad preescolar

My Talking Computer, de Microspeech, está dirigido a niños a partir de tres años de edad, y la empresa afirma que su máquina ya se está utilizando en escuelas primarias de Gran Bretaña. El "ordenador" está alojado en una carcasa de plástico ligera que mide 23×25 cm. En un panel sensible al tacto situado en la parte delantera de la máquina se colocan cubiertas para los diversos programas, que se mantienen en su sitio mediante un marco. En este sentido se parece al teclado Touchmaster, aunque My Talking Computer tiene muchos menos puntos de contacto. Lleva instalado My Talking Clock que, utilizado junto con uno de los programas incorporados, enseña al niño a decir la hora.

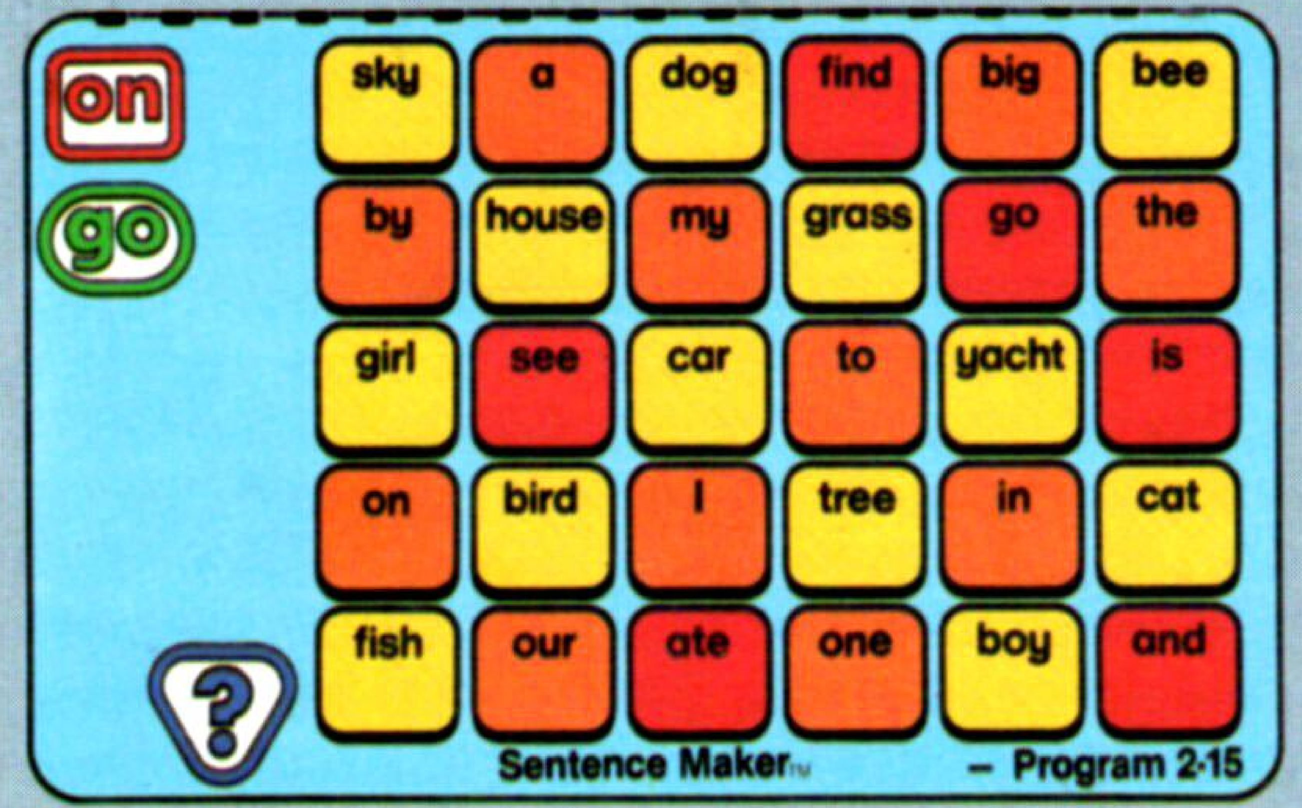
Al costado de la máquina hay una ranura en la cual se insertan los módulos de los programas (dispositivos similares a cartuchos). Aunque My Talking Computer representa un enfoque nuevo e ingenioso para enseñar a niños pequeños, los programas residentes están obviamente limitados en cuanto a sus aplicaciones, lo cual acorta de manera inevitable la vida útil de la máquina en cuanto concierne al pequeño. No obstante, los fabricantes pretenden superar este problema proporcionando una ranura de ampliación que permite la instalación de ROM extras. A pesar de que la primera de estas ROM, Expansion Module One (módulo de ampliación uno), parece ser de un nivel apenas ligeramente superior al de la ROM existente, la intención del fabricante parece ser producir una serie que vaya sofisticándose a medida que el niño crezca. Encima de la ranura de ampliación hay un pequeño altavoz. La máquina opera mediante cinco pilas de 1,5 V o bien mediante alimentación eléctrica externa.

Los programas están almacenados en un único chip de ROM que, según Microspeech, contiene más de 120 K de datos, si bien el ordenador no cargará esta información de una sola vez. El software basado en ROM está dividido en cinco programas maestros, cada uno de los cuales está subdividido en varios otros basados en el mismo tema. El primer programa maestro está dedicado a enseñar aritmética, con ejercicios tales como reconocimiento de números y operaciones sencillas de suma, resta, multiplicación y división. El segundo enseña la relación entre imágenes y palabras. El tercero es una "calculadora parlante" que lee las cifras a medida que el niño las va entrando y luego dice el resultado. El cuarto programa consiste en juegos "parlantes" que comprueban la habilidad del pequeño para responder a las relaciones, expresadas en términos tales como "hallar el perro". El quinto programa, que ya hemos mencionado, es para aprender a decir la hora.

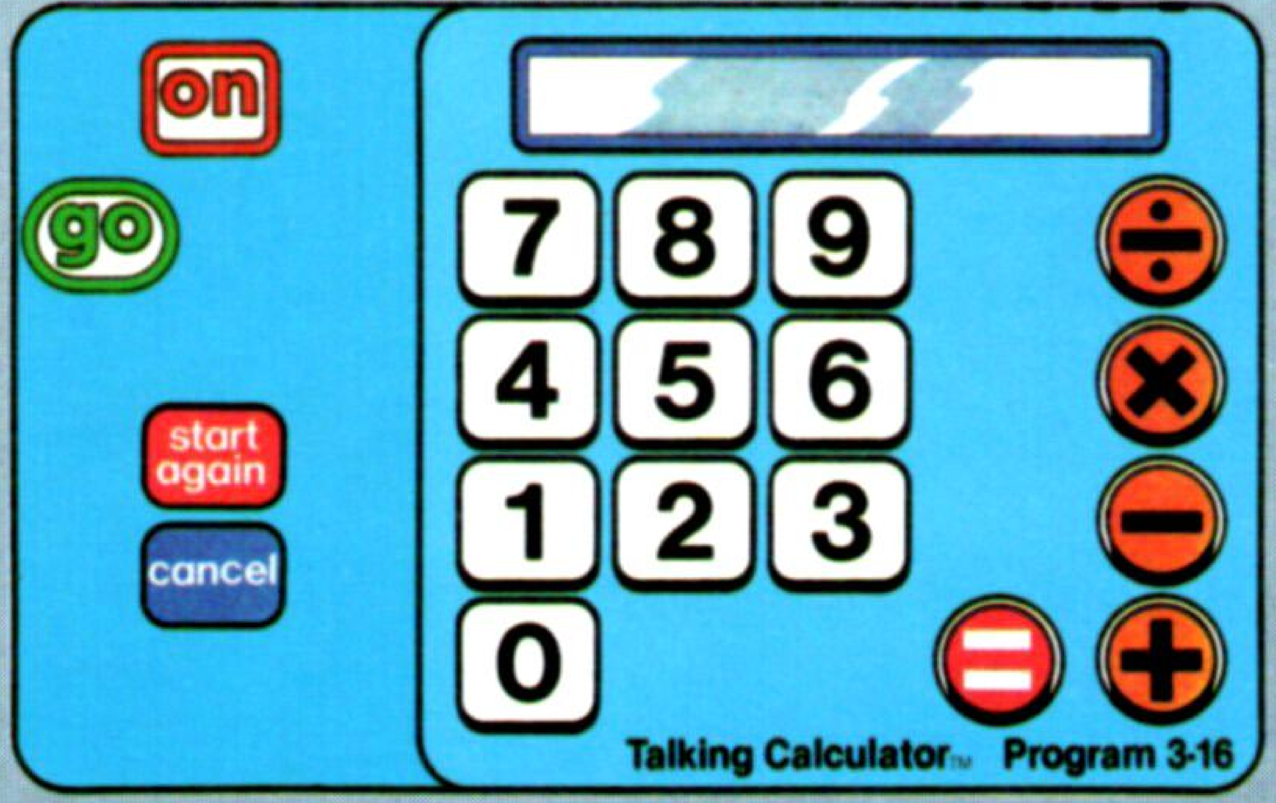
El niño utiliza un programa determinado remitiéndose a un libro de cubiertas con anillas e instalando la cubierta seleccionada debajo del marco situado en la parte frontal del ordenador. Pulsando el cuadrado ON de la cubierta la máquina se enciende y una voz femenina responde diciendo "Hello".



Talking Story
Esta cubierta está diseñada para niños de 2 a 12 años. El ordenador reconoce qué programa se está utilizando en función de las posiciones de los "botones" ON y GO. Cuando se pulsa un cuadrado con palabra, el ordenador la pronuncia



Sentence Maker
Este programa enseña a reconocer palabras. Cuando el niño pulsa el signo de interrogación, el ordenador dice una palabra y el niño debe pulsar la tecla correspondiente a la misma



Talking Calculator
Es uno de los programas más interesantes de los que dispone My Talking Computer. Pulsando los números y los signos aritméticos se pueden calcular sumas sencillas



Ian McKinnell

Decir la hora
El ordenador lleva instalado My Talking Clock (mi reloj parlante) encima del teclado al tacto. En la cara inferior del reloj hay un conjunto de protuberancias, que corresponden al teclado de los ordenadores. Las manecillas del reloj se pueden mover alrededor del cuadrante y el micro dirá qué hora se está visualizando

La máquina le pide entonces al niño que pulse la tecla GO de la cubierta. El ordenador "sabe" cuál es el programa que se ha seleccionado porque las teclas ON y GO están situadas en distintas posiciones en cada cubierta. Este método de operación es especialmente útil para los niños de corta edad.

Una vez que el niño ha aprendido que ON hace que el ordenador se encienda y que GO da comienzo al programa, puede luego cargar y ejecutar cualquiera de los programas sin que para ello se requiera la ayuda de un adulto. Las cubiertas están revestidas en plástico, lo que significa que las eventuales manchas de mermelada y chocolate se podrán limpiar.

Para evaluar el valor educativo de un elemento de hardware o de software se han de aplicar ciertos criterios. El paquete debe enseñar aquello que pretende, por ejemplo, y no debe presuponer ningún conocimiento previo del tema que intenta impartir: las páginas de instrucciones escritas son evidentemente inaceptables en un programa para "aprender a leer". También sirve de ayuda el hecho de que sea entretenido y fácil de utilizar.

El sintetizador de voz

El sintetizador de voz incorporado es lo mejor del ordenador. De hecho, hace que uno se pregunte por qué las calculadoras no poseen facilidades de voz incorporadas. Así y todo, tiene sus limitaciones. El habla está almacenada como palabras enteras, a las que se accede luego individualmente para construir las frases utilizadas. La enfática pronunciación resultante en realidad no se puede evitar, puesto que las mismas palabras se aplican en numerosos contextos. La voz femenina utilizada posee acento norteamericano, lo que tal vez induzca a que los padres británicos se quejen de que se les enseñe a sus hijos a hablar en "norteamericano", pero ello molestará más a los adultos que a los niños. El problema más grave, tratándose de una máquina diseñada para aplicaciones educativas, es que muchos de los vocablos no son suficientemente claros: es difícil, por ejemplo, diferenciar las palabras *"by", "my"* y *"sky"*. Incluso tras la repetición de ciertas frases resulta difícil determinar lo que se ha dicho.

Ahora podemos juzgar si My Talking Computer satisface los criterios que hemos definido como esenciales en todo medio auxiliar educativo. La facilidad con la cual se pueden instalar y ejecutar nuevos programas es ciertamente digna de elogio. El hecho de que el ordenador hable es una enorme ventaja para enseñar a un niño a leer, y evita hábilmente la necesidad de complicadas páginas de instrucciones. A pesar de las dudas sobre la calidad de la voz, la capacidad del ordenador para asociar el sonido de la palabra con su forma escrita es un método de enseñanza mucho más directo que el que ofrecen los paquetes de software, que simplemente asocian la palabra con imágenes; aunque, por supuesto, My Talking Computer también lo hace. Su capacidad para responder verbalmente asegura, asimismo, que su utilización sea entretenida.

My Talking Computer no es, como es obvio, un ordenador en el sentido literal de la palabra, puesto que el usuario no lo puede "programar". Sin embargo, demuestra la aplicación de la tecnología del microordenador en el campo de la educación.

MY TALKING COMPUTER

DIMENSIONES
250×230×70 mm

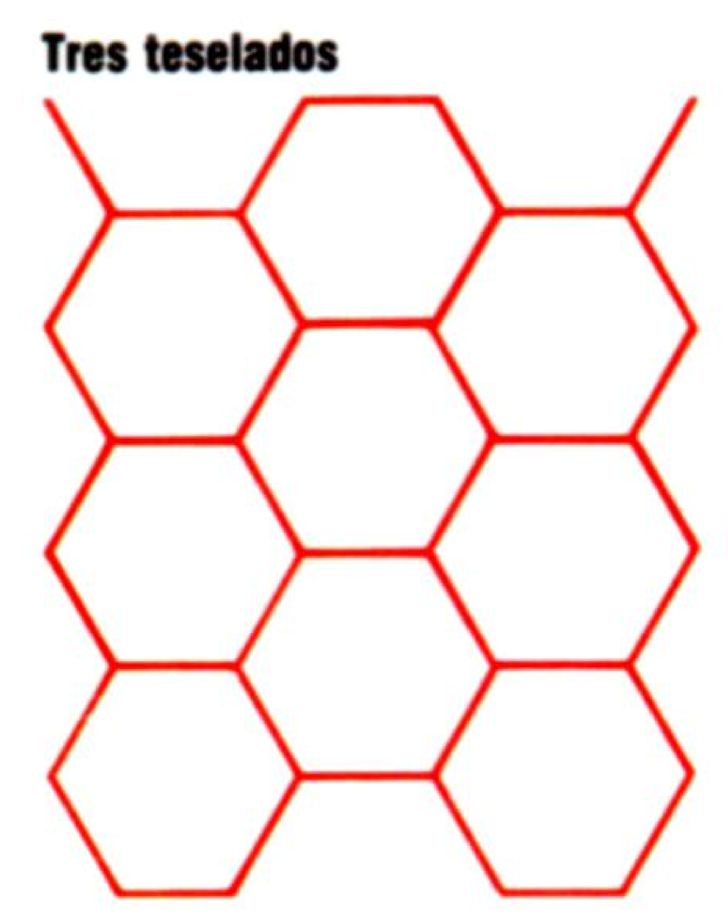
SOFTWARE
El software se proporciona en la ROM del ordenador. Se ofrece software extra en forma del Expansion Module 1, que se enchufa en una ranura para cartuchos que posee la máquina

VENTAJAS
Es sumamente fácil de utilizar. La sencillez del procedimiento de carga y la facilidad para síntesis de voz significan que el niño no necesita saber leer para poder utilizar la máquina

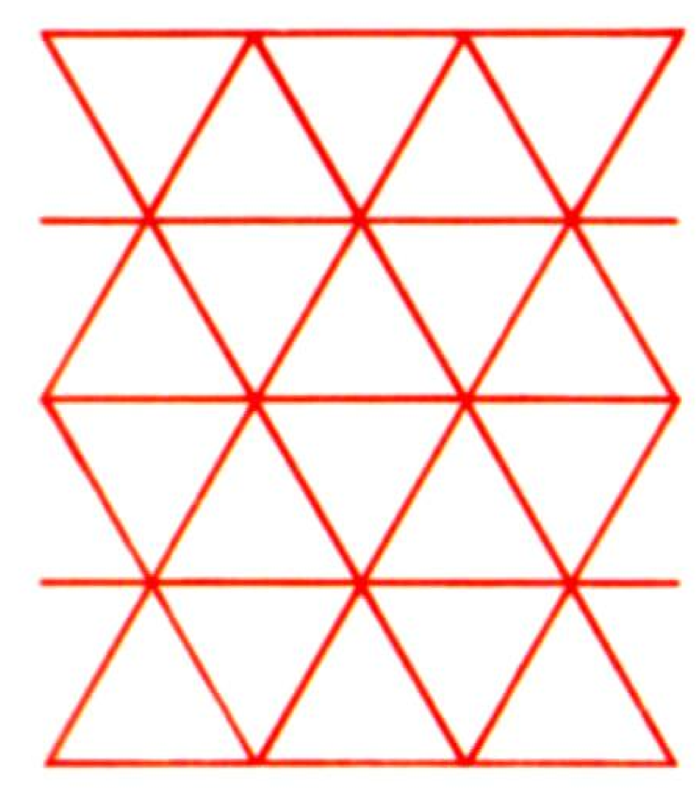
DESVENTAJAS
Con frecuencia el habla es enfática y poco clara. La máquina no es programable y no se puede considerar un auténtico microordenador

Líneas finales

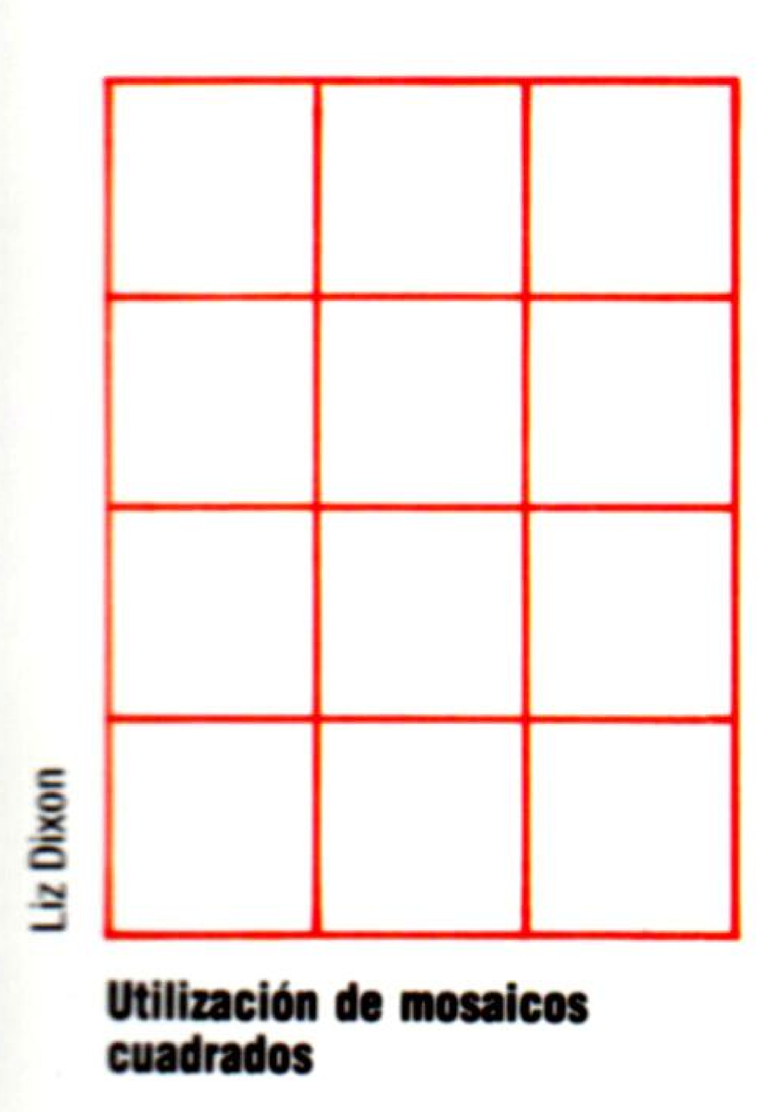
Tres teselados



Utilización de mosaicos hexagonales



Utilización de mosaicos de triángulo equilátero



Utilización de mosaicos cuadrados

Liz Dixon

En este último capítulo de nuestro curso veremos cómo se utiliza el LOGO para teselar diversas y atractivas formas geométricas

Un *teselado* cubre la superficie de un plano mediante la unión de mosaicos de la misma forma, sin dejar espacios entre las piezas individuales.

Algunos polígonos regulares formarán teselados: cuadrados, triángulos equiláteros y hexágonos (como vemos en el margen izquierdo). Sin embargo, ninguno de los otros polígonos se puede teselar. Para ver por qué, tomemos un ejemplo: un pentágono. El ángulo del vértice de un pentágono regular es de 108°. Coloque tres de ellos juntos en un punto focal y habrá empleado sólo 324°, de modo que queda un vacío. Añada un cuarto y el ángulo será entonces de 432°, lo que significa que las formas se superponen. Tanto los cuadrados como los triángulos equiláteros y los hexágonos se teselan precisamente porque los ángulos de sus vértices (90°, 60° y 120° respectivamente) dividen 360 un número exacto de veces.

Escribir procedimientos en LOGO para producir estos teselados es bastante directo. Probablemente el mejor enfoque sea escribir un procedimiento para dibujar el motivo empezando desde el centro y retornando al centro (ello supone una transparencia de estado). Luego podemos utilizarlo dentro de un "superprocedimiento" que desplace la tortuga hasta el centro de la siguiente forma a dibujar en cada ocasión.

Los otros procedimientos dibujan un teselado simple de cuadrados. Los teselados de triángulos equiláteros y hexágonos se pueden dibujar de forma similar.

```
TO TES
   PENUP SETXY (–100)(90) PENDOWN
   REPEAT 5[LINEAVERT SETXY (–100)
   (YCOR–40)
END

TO LINEAVERT
   REPEAT 5[CUAD 40 SETXY XCOR+40]
END

TO CUAD :S
   PENUP LEFT 45
   FORWARD :S*(SQRT 2)/2
   RIGHT 135 PENDOWN
   FORWARD :S RIGHT 90 FORWARD :S RIGHT 90
   FORWARD :S RIGHT 90 FORWARD :S RIGHT 90
   PENUP LEFT 135
   BACK :S*(SQRT2)/2 RIGHT 45
END
```

Se pueden componer teselados a partir de motivos más complejos que los polígonos regulares y también se pueden realizar utilizando una mezcla de formas. No obstante, los teselados basados en polígonos regulares poseen un inmenso potencial. Muchos de los dibujos de M.C. Escher son variaciones de teselados regulares simples.

Un método directo para construir diagramas más complejos implica modificar CUADRADO reemplazando las instrucciones FORWARD que dibujan los lados del cuadrado por procedimientos. La única regla que debemos observar es que cualquier modificación en el lado superior del cuadrado se debe compensar con otra en la base, y cualquier modificación en el lado derecho se ha de compensar con otra en el lado izquierdo, de modo que las formas sigan encajando.

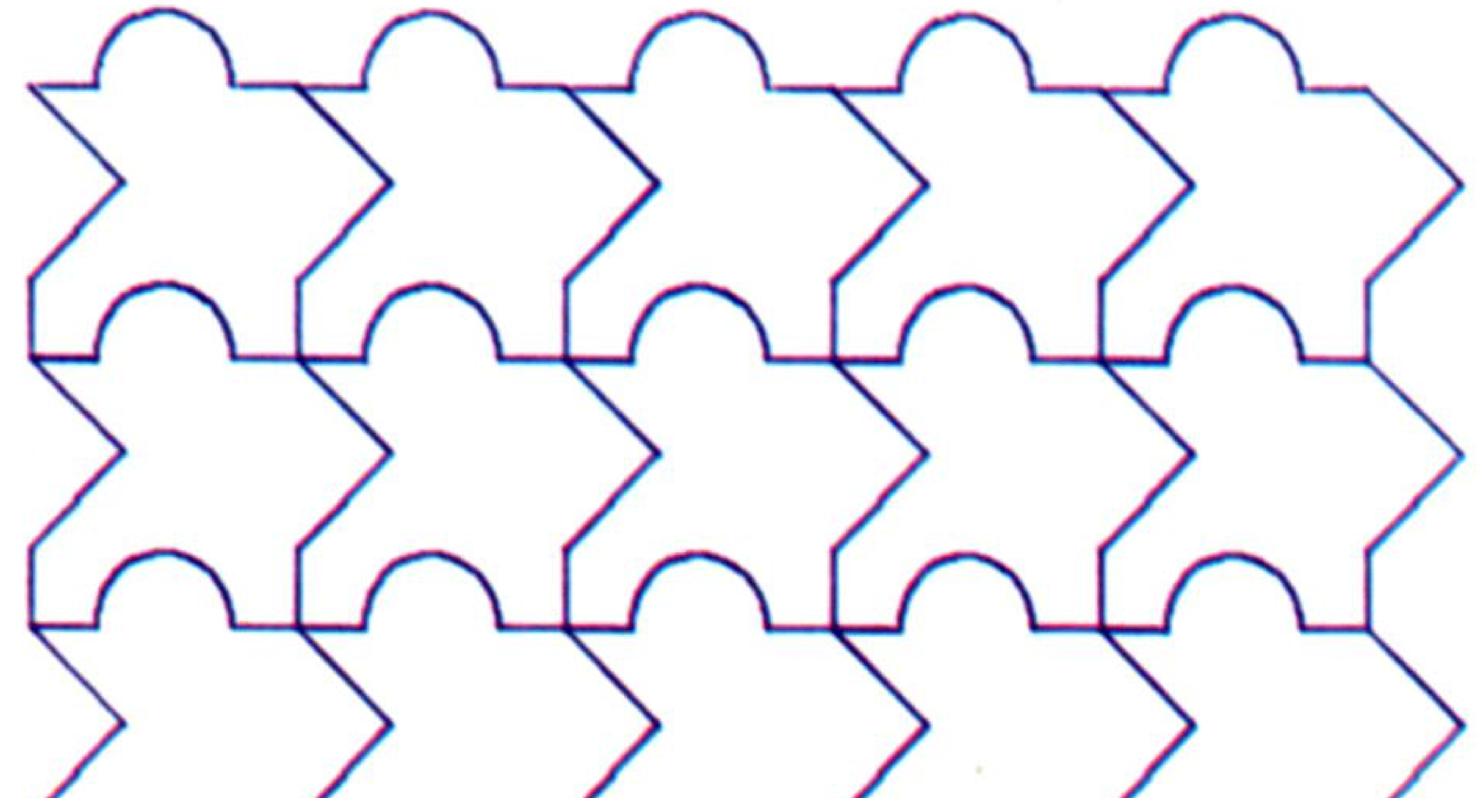
Para dibujar la forma básica, sencillamente reemplazamos las instrucciones que dibujan los lados del cuadrado por procedimientos para dibujar las líneas nuevas. El nuevo teselado (que vemos abajo) se dibuja entonces volviendo a llamar a TES. Éste es el listado completo:

```
TO CUAD :S
   PENUP LEFT 45
   FORWARD :S*(SQRT 2)/2
   RIGHT 135 PENDOWN
   LADOA :S RIGHT 90 LADOB :S
   RIGHT 90
   LADOC :S RIGHT 90 LADOD :S
   RIGHT 90
   PENUP LEFT 135
   BACK :S*(SQRT 2)/2
   RIGHT 45
END

TO LADOA :S
   FORWARD :S/4 LEFT 90
   REPEAT 19[FORWARD 2*3.1416* :S/144
   RIGHT 10]
   LEFT 100 FORWARD :S/4
END

TO LADOB :S
   LEFT 45 FORWARD :S/2
   RIGHT 90 FORWARD :S/2
   LEFT 45 FORWARD :S–:S*(SQRT 2)/2
END

TO LADOC :S
   FORWARD :S/4 RIGHT 90
   REPEAT 19 [FORWARD 2*3.1416*:S/144 LEFT10]
   RIGHT 100 FORWARD :S/4
END
```





```
TO LADOD :S
   FORWARD :S−:S*(SQRT 2)/2 RIGHT 45
   FORWARD :S/2 LEFT 90
   FORWARD :S/2 RIGHT 45
END
```

La técnica de Escher era, por supuesto, más complicada, puesto que él también modificaba lentamente las formas a medida que se iban "moviendo" a través de su teselado. Sería interesante un procedimiento en LOGO que hiciera lo mismo...

Usando pegamento

Una forma bastante diferente de enfocar la cuestión de los teselados es la que ofrecen Harold Abelson y Andrea diSessa a modo de ejercicio en su libro *Turtle geometry*. Su enfoque es así: imagine una forma dibujada en el interior de un cuadrado, por ejemplo, el patrón de un mosaico, y después pegue cuatro de éstos entre sí para formar un cuadrado más grande. Tome este cuadrado más grande y pegue cuatro de éstos entre sí, y así sucesivamente.

Existen muchos métodos diferentes de pegar. De hecho, puesto que cada una de las cuatro piezas se puede orientar en una de cuatro direcciones, hay 4×4×4×4, o 256, posibilidades. Los patrones resultantes son producto tanto de lo que se había dibujado en el mosaico como del método de pegado empleado. Nuestro procedimiento, MARCA, dibuja el motivo básico que vemos a la derecha.

```
TO MARCA :S
   PENDOWN LEFT 45
   FORWARD :S*(SQRT 2)/2
   BACK :S*(SQRT 2)/2
   LEFT 90 FORWARD :S*(SQRT 2)/2 PENUP
   BACK :S*(SQRT 2) PENDOWN
   FORWARD :S*(SQRT 2)/4 LEFT 45
   FORWARD :S/2 LEFT 45
   FORWARD :S*(SQRT 2)/4 PENUP
   BACK :S*(SQRT 2)/2 LEFT 135
END
```

Figuras en evolución
Las figuras de teselado son formas que encajan entre sí exactamente para cubrir un plano. *Metamorfosis II*, de M. C. Escher (1898-1972), en el cual está basado nuestro gráfico creado por ordenador, demuestra cómo una sencilla figura teselada, como un hexágono regular, se puede ir modificando progresivamente para producir un teselado más complejo; en este caso, la forma de una sofisticada lagartija. La técnica de Escher pone de manifiesto el cambio paulatino de lo simple a lo complejo a medida que el ojo va avanzando por la escena

Ian McKinnell

Un posible procedimiento de pegado sería PEGAR1. Para ejecutarlo, digite DRAW PENUP PEGAR1 [MARCA]100. Podemos ver el patrón resultante.

```
TO PEGAR1 :PROC :S
   PIEZA.CUADRADO 0 :PROC :S
   PIEZA.CUADRADO 0 :PROC :S
   PIEZA.CUADRADO 0 :PROC :S
   PIEZA.CUADRADO 0 :PROC :S
END
```



PIEZA.CUADRADO dibuja uno de los cuatro cuadrados que componen el cuadrado más grande. Toma tres entradas: :A determina la orientación del mismo, :PROC es el nombre del procedimiento para dibujar el motivo y :S es la longitud del lado. No es factible, dados los límites de la pantalla, mantener en continua ampliación el tamaño de los lados, de modo que el procedimiento toma el tamaño del cuadrado exterior como entrada y calcula el tamaño de los cuadrados más pequeños.

```
TO PIEZA.CUADRADO :A :PROC :S
   FORWARD :S / 4 RIGHT 90
   FORWARD :S / 4 RIGHT 90*:A
   RUN SENTENCE :PROC :S / 2 LEFT 90*:A
   BACK :S / 4 LEFT 90
   BACK :S / 4 RIGHT 90
END
```

He aquí algunos otros posibles métodos de pegado:

```
TO PEGAR2 :PROC :S
   PIEZA.CUADRADO 0 :PROC :S
   PIEZA.CUADRADO 1 :PROC :S
   PIEZA.CUADRADO 2 :PROC :S
   PIEZA.CUADRADO 3 :PROC :S
END

TO PEGAR3 :PROC :S
   PIEZA.CUADRADO 0 :PROC :S
   PIEZA.CUADRADO 2 :PROC :S
   PIEZA.CUADRADO 3 :PROC :S
   PIEZA.CUADRADO 1 :PROC :S
END

TO PEGAR4 :PROC :S
   PIEZA.CUADRADO 3 :PROC :S
   PIEZA.CUADRADO 2 :PROC :S
   PIEZA.CUADRADO 1 :PROC :S
   PIEZA.CUADRADO 0 :PROC :S
END

TO PEGAR5 :PROC :S
   PIEZA.CUADRADO 2 :PROC :S
   PIEZA.CUADRADO 1 :PROC :S
   PIEZA.CUADRADO 0 :PROC :S
   PIEZA.CUADRADO 3 :PROC :S
END
```

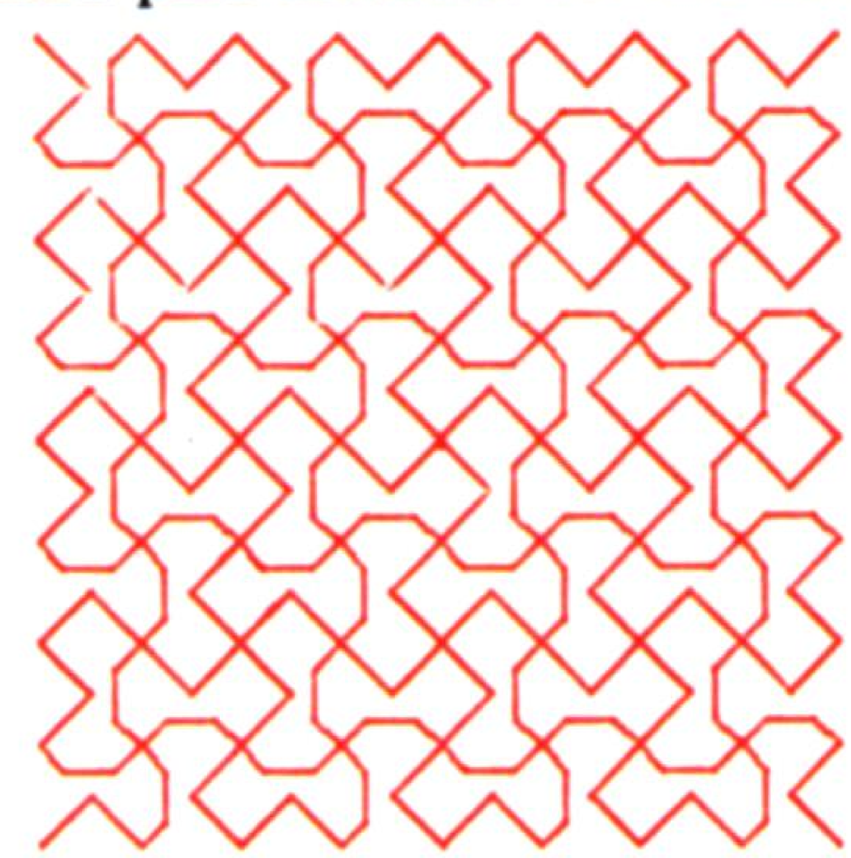
MARCA es el nombre de la instrucción para dibujar la forma que requiere una entrada. Ésta la pegamos con PEGAR1 [MARCA] 100. Observando esta línea de instrucciones podemos ver que PEGAR1 [MARCA] se puede considerar como una instrucción que sólo exige un número con el objeto de dibujar una forma. Podemos, por lo tanto, utilizar [PEGAR1 [MARCA]] como la entrada para otra instrucción PEGAR. Por ejemplo:

```
PEGAR2 [PEGAR1 [MARCA]] 100
```

¡La diversión apenas empieza! Considere:

```
PEGAR3 [PEGAR2[PEGAR1[MARCA]]]100
```

Ahora tenemos 256×256×256 combinaciones de pegado a tres niveles para considerar. ¡Pero tampoco hay razón para detenerse en tres niveles.

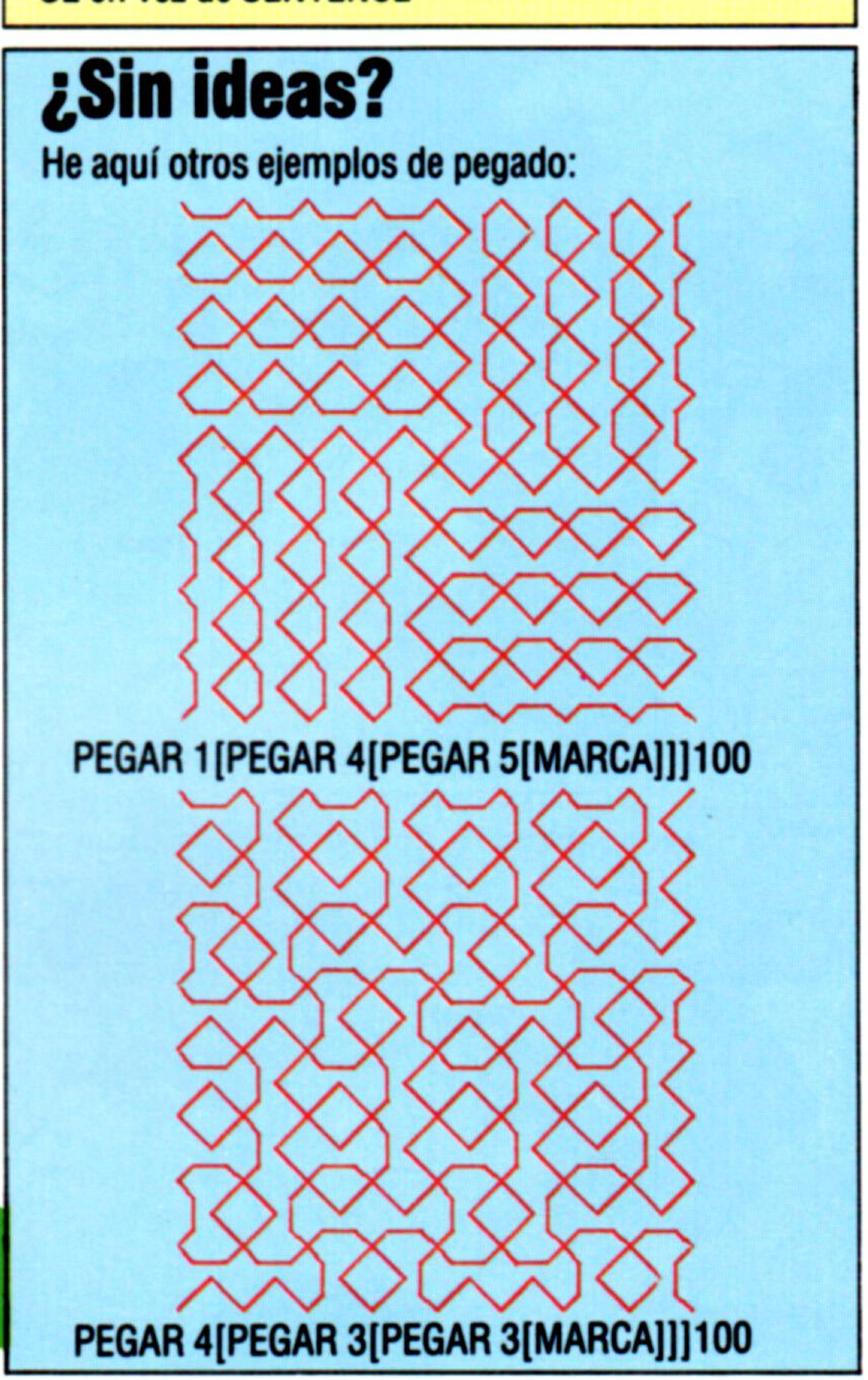


Complementos al LOGO

En todas las versiones LCSI, utilice SETPOS en vez de SETXY, y recuerde que ha de ir seguida de una lista. En el LOGO Atari debe emplear la abreviatura SE en vez de SENTENCE

¿Sin ideas?

He aquí otros ejemplos de pegado:



PEGAR 1[PEGAR 4[PEGAR 5[MARCA]]]100

PEGAR 4[PEGAR 3[PEGAR 3[MARCA]]]100

Misión de reconocimiento

Diseñaremos software para que el robot explore y visualice una superficie determinada

Para lograr que nuestro robot explore un área dada, debemos primero concebir un patrón de exploración que cubra la zona. Podríamos empezar por considerar un patrón de exploración en el cual el robot se mueva hacia atrás y hacia adelante a través de la superficie sondeando en busca de algún objeto. Al localizar lo podría optar por tantear alrededor de sus lados antes de proceder a "barrer" el resto de la superficie. Sin embargo, esta clase de algoritmo es difícil de programar y puede plantear problemas si dentro de la zona a explorar hay más de un objeto. Un curso de acción alternativo al encontrar el objeto es dar un rodeo y continuar explorando como antes. Los pasos de un algoritmo de esta clase se pueden describir así:

```
REPEAT
   REPEAT
   Sondear hacia adelante en busca de un objeto
   UNTIL llegar al borde del área definida
   o hallar un objeto
Avanzar un "ancho de franja"
Girar
UNTIL llegar al borde del área definida
```

Al completar esta exploración horizontal, el robot no habrá sondeado toda el área disponible si en la misma hubiera algún objeto presente. Utilizando este patrón de exploración, la superficie por detrás de cualquier objeto ("la zona ciega") quedará sin explorar. Se requiere al menos una exploración vertical (es decir, a 90° de la primera).

Para que el robot pueda llevar a cabo una exploración satisfactoria es necesario introducir una mínima modificación en los dos sensores delanteros. Debido a que las ruedas del robot sobresalen por los lados, existe el peligro de que entren en contacto con un objeto mientras el robot pasa explorando junto a su lado. Para subsanar este inconveniente hemos de añadir un escudo adelante de cada sensor para proteger las ruedas. Cada escudo ha de ser un rectángulo de aproximadamente 95×25 mm, de un material ligero (plástico duro o metal). Asegúrese de que el material empleado no sea tan pesado como para empujar el sensor hacia abajo cuando esté en posición cerrado. Los escudos se deben montar encima de los sensores utilizando cinta adhesiva o cola, y se deben colocar de modo que se extiendan hacia afuera para cubrir exactamente las ruedas. Compruebe la acción del interruptor del sensor para asegurar que se pueda abrir y cerrar.

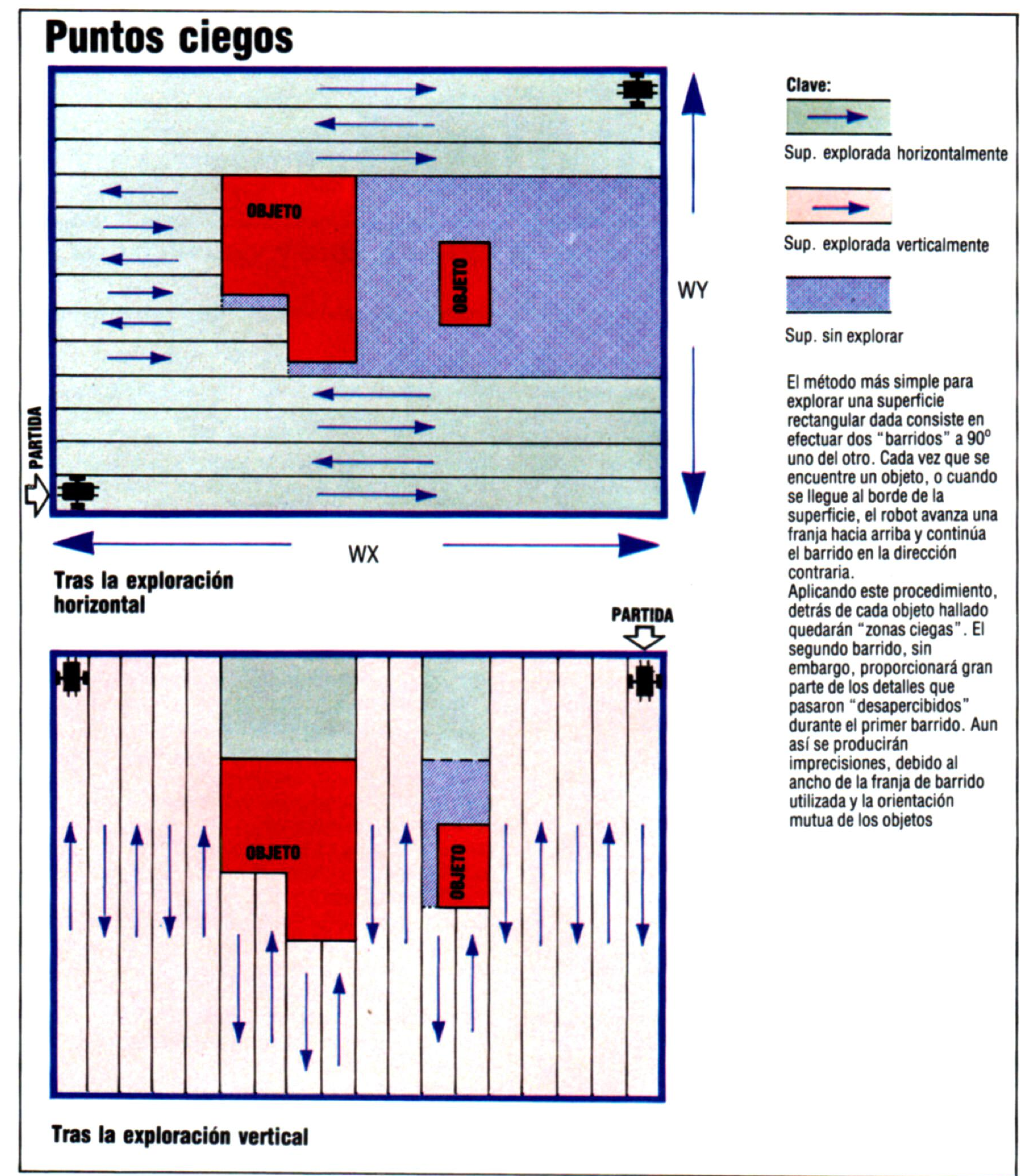
El programa utilizado para hacer que el robot efectúe una exploración de superficie se ha escrito empleando un único patrón de exploración horizontal y vertical. A medida que el robot va completando su recorrido de cada franja de la exploración, se va rellenando con color la zona correspondiente en la pantalla del ordenador. Finalizada la exploración, se visualizará en la pantalla una representación pictórica de la superficie, tal como la "ve" el robot.

Las dos versiones de nuestro programa piden al usuario que entre las dimensiones de la superficie a explorar. Utilizando la Mode 4 del BBC o la visualización en alta resolución del Commodore 64, se hace que cada pixel de pantalla corresponda a un cuadrado de 4 mm de la superficie que se esté explorando. Las dimensiones máximas en las direcciones horizontal y vertical son, por consiguiente, 1 279 y 1 023 mm, respectivamente, para ambas máquinas. Observe que cada programa utiliza un ancho de franja de 40 mm.

A medida que el robot, durante la exploración, va completando cada una de las franjas, la visualización gráfica se va actualizando utilizando los procedimientos (o subrutinas) denominados XPLOT e YPLOT. En la versión para el BBC Micro, la superficie de franjas de la pantalla se rellena empleando las instrucciones MOVE y DRAW para producir una serie de líneas paralelas adyacentes. La versión para el Commodore 64, sin embargo, se basa en las dos rutinas en código máquina (PLOTSUB y LINESUB) diseñadas previamente en el curso para compensar la falta de instrucciones para dibujar en alta resolución del BASIC Commodore. Usted debe de tener copia de los archivos objeto finales para estas dos rutinas, que se pueden cargar utilizando las líneas 30 y 40 de la versión Commodore del programa. Por otra parte, tendrá cargadores en BASIC para estas dos rutinas. Cada rutina se debe cargar y ejecutar por separado antes de cargar y ejecutar el programa. En este último caso, las líneas 30 y 40 se pueden omitir.

Además de la zona ciega sustancial que podría quedar tras dos exploraciones, se pueden presentar otras dificultades adicionales. Entre ellas, la más destacable es que el programa trabaja con niveles de éxito variables para formas u orientaciones diferentes. Tendrá el mayor éxito con formas rectangulares, orientadas a lo largo de los ejes horizontal y vertical. Las triangulares y circulares producirán zonas ciegas más grandes, dado que el patrón de exploración tenderá a simplificar las irregularidades de los objetos que se encuentren. Quizá desee, a modo de ejercicio, ampliar el programa que ofrecemos aquí de modo que se efectúe otra exploración vertical. El robot puede, asimismo, hallarse en dificultades si se encuentra con un objeto durante su maniobra para comenzar una franja nueva.

Puntos ciegos



El método más simple para explorar una superficie rectangular dada consiste en efectuar dos "barridos" a 90° uno del otro. Cada vez que se encuentre un objeto, o cuando se llegue al borde de la superficie, el robot avanza una franja hacia arriba y continúa el barrido en la dirección contraria.
Aplicando este procedimiento, detrás de cada objeto hallado quedarán "zonas ciegas". El segundo barrido, sin embargo, proporcionará gran parte de los detalles que pasaron "desapercibidos" durante el primer barrido. Aun así se producirán imprecisiones, debido al ancho de la franja de barrido utilizada y la orientación mutua de los objetos

Kevin Jones

Listado para el BBC Micro

```
10 REM **** EXPLORADOR DE FORMAS BBC ****
20 MODE 4
30 PROCinicializar
40 PROCexplorar_dimensiones
50 PROCexplor_horiz
60 PROCexplor_vert
70 END
80 DEF PROCexplor_horiz
90 objetivo=wx:sentido=izquierda:xorigen=0
100 REPEAT
110 REPEAT
120 PROCmover(adelante,dx):x=x+dx
130 UNTIL(?REGDAT AND 192)<>ningun_parachoques OR x=objetivo
140 PROCxplot
150 dx=-dx:y=y+dy:xorigen=x
160 IF objetivo=0 THEN objectivo=wx ELSE objetivo=0
170 IF y<wy THEN PROCsig_franja(sentido)
180 IF sentido=derecha THEN sentido=izquierda ELSE sentido=derecha
190 UNTIL y>wy
200 ENDPROC
210 :
220 DEF PROCexplor_vert
230 IF x=0 THEN sentido=izquierda:dx=anchura ELSE sentido=derecha:dx=-anchura
```

Listado para el Commodore 64

```
10 REM **** EXPLORADOR DE FORMAS CBM ****
20 DN=8:REM SI CASS DN=1
30 IF A=0 THEN A=1:LOAD"PLOTSUB.HEX",DN,1
40 IF A=1 THEN A=2:LOAD"LINESUB.HEX",DN,1
50 GOSUB1000:REM INICIALIZAR
60 GOSUB2000:REM EXPLORAR DIMENSIONES
70 GOSUB5000:REM PASAR A MODALIDAD ALTA RES
80 GOSUB3000:REM EXPLORACION HORIZONTAL
90 GOSUB4000:REM EXPLORACION VERTICAL
100 GOSUB5100:REM SALIR DE MODALIDAD ALTA RES
110 END
120 :
1000 REM **** S/R INICIALIZAR ****
1010 RDD=56579:REGDAT=56577
1020 POKE RDD,15:POKE REGDAT,1
1030 AD=4:AT=2:IZ=6:DE=0
1040 ID=3.34446:IA=375/90
1050 RB=128:LB=64:BB=0:NB=192
1060 WD=40:DW=WD/10:X=0:Y=0:DX=DW:DY=WD
1070 REM ** DIRECCIONES DE COMIENZO C/M **
1080 HIRES=49422:LINESUB=49934
1090 RETURN
1100 :
```

Continuación del listado para el BBC Micro

```
240 dy=-dw:objetivo=0:yorigen=y
250 PROCgirar(sentido,90)
260 REPEAT
270 REPEAT
280 PROCmover(adelante,dy):y=y+dy
290 UNTIL(?REGDAT AND 192)<>ningun_parachoques OR y=objetivo
300 PROCyplot
310 dy=-dy:x=x+dx:yorigen=y
320 IF objetivo=0 THEN objetivo=wy ELSE objetivo=0
330 PROCsig_franja(sentido)
340 IF sentido=derecha THEN sentido=izquierda ELSE sentido=derecha
350 UNTIL x>wx OR x<0
360 ENDPROC
370 :
380 DEF PROCsig_franja(rumbo)
390 PROCmover(atras,30)
400 PROCgirar(rumbo,90)
410 PROCmover(adelante,anchura)
420 PROCgirar(rumbo,90)
430 ENDPROC
440 :
450 DEF PROCexplorar_dimensiones
460 INPUT"DIMENSION X EN MM";wx
470 IF wx>1279 THEN 460
480 INPUT"DIMENSION Y EN MM";wy
490 IF wy>1023 THEN 480
500 wx=anchura*(wx DIV anchura)
510 wy=anchura*(wy DIV anchura)
520 CLS
530 ENDPROC
540 :
550 DEF PROCinicializar
560 RDD=&FE62:REGDAT=&FE60
570 ?RDD=15:REM LINEAS 0-3 SALIDA
580 ?REGDAT=1:REM ENCENDER BIT REINICIO
590 adelante=4:atras=2:izquierda=6:derecha=0
600 coef_id=3.34446:coef_ia=375/90
610 parachoques_derecho=128:parachoques_izquierdo=64
620 ambos_parachoques=0:ningun_parachoques=192
630 anchura=40:dw=anchura/10
640 x=0:y=0:dx=dw:dy=anchura
650 MOVE 0,0
660 ENDPROC
670 :
680 DEF PROCmover(dir,distancia)
690 ?REGDAT(?REGDAT AND 1)OR dir
700 impulsos=coef_id*ABS(distancia)
710 FOR I=1 TO impulsos:PROCimpulso:NEXT I
720 ENDPROC
730 :
740 DEF PROCgirar(dir,angulo)
750 ?REGDAT=(?REGDAT AND 1)OR dir
760 impulsos=coef_ia*angulo
770 FOR I=1 TO impulsos:PROCimpulso:NEXT I
780 ENDPROC
790 :
800 DEF PROCimpulso
810 ?REGDAT=(?REGDAT OR 8)
820 ?REGDAT=(?REGDAT AND 247)
830 ENDPROC
840 DEF PROCxplot
850 FOR I=0 TO anchura STEP 4
860 MOVExorigen,y+I
870 DRAWx,y+I
880 NEXT I
890 ENDPROC
900 :
910 DEF PROCyplot
920 FOR I=0 TO anchura STEP 4
930 MOVE x+I,yorigen
940 DRAWx+I,y
950 NEXT I
960 ENDPROC
```

Continuación del listado para el Commodore 64

```
2000 REM **** DIMENSIONES DE EXPLORACION ****
2010 INPUT"DIMENSION X EN MM";WX
2020 IF WX/4>319 THEN 2010
2030 INPUT"DIMENSION Y EN MM";WY
2040 IF WY/4>199 THEN 2030
2050 WX=WD*INT(WX/WD):WY=WD*INT(WY/WD)
2060 RETURN
2070 :
3000 REM **** EXPLORACION HORIZONTAL ****
3010 TG=WX:SE=LF:XS=0
3020 DR=FW:DS=DX:GOSUB7000:REM MOVER
3030 X=X+DX
3040 IF(PEEK(REGDAT)AND 192)=NB AND X<>TG THEN 3020
3050 GOSUB 9000:REM XPLOT
3060 DX=-DX:Y=Y+DY:XS=S
3070 IF TG=0 THEN TG=WX:GOTO 3090
3080 TG=0
3090 IF Y<WY THEN WA=SE:GOSUB8000:REM SIGUIENTE FRANJA
3100 IF SE=RT THEN SE=LF:GOTO 3120
3110 SE=RT
3120 IF Y<WY THEN 3020
3130 RETURN
3140 :
4000 REM **** EXPLORACION VERTICAL ****
4010 IF X=0 THEN SE=LF:DX=WD:GOTO 4030
4020 SE=RT:DX=-WD
4030 DY=-DY:TG=0:YS=Y
4040 DR=SE:AG=90:GOSUB7100:REM GIRAR
4050 DR=FW:DS=DY:GOSUB7000:REM MOVER
4060 Y=Y+DY
4070 IF(PEEK(REGDAT)AND192)=NB AND Y<>TG THEN 4050
4080 GOSUB9100:REM YPLOT
4090 DY=-DY:X=X+DX:YS=Y
4100 IF TG=0 THEN TG=WY:GOTO 4120
4110 TG=0
4120 WA=SE:GOSUB8000:REM SIGUIENTE FRANJA
4130 IF SE=RT THEN SE=LF:GOTO 4150
4140 SE=RT
4150 IF X<WX AND X>0 THEN 4050:REM REPETIR
4160 RETURN
4170 :
5000 REM **** ENTRAR ALTA RES ****
5010 POKE 49408,1:POKE 49409,1
5020 POKE 49410,1:SYS HIRES:RETURN
5030 :
5100 REM **** SALIR DE ALTA RES ****
5110 POKE 49408,0:POKE 49409,0
5120 POKE 49410,1:SYS HIRES:RETURN
5130 :
6000 REM **** ENTRAR LINESUB ****
6010 MHI=INT(X1/256):MLO=X1-256*MHI
6020 NHI=INT(X2/256):NLO=X2-256*NHI
6030 POKE 49920,MLO:POKE 49921,MHI
6040 POKE 49922,NLO:POKE 49923,NHI
6050 POKE 49924,Y1:POKE 49925,Y2
6060 SYS LINESUB:RETURN
6070 :
7000 REM **** MOVER (DR,DS) ****
7010 POKE REGDAT,(PEEK(REGDAT)AND 1)OR DR
7020 PL=PD*DS
7030 FOR I=1 TO PL:GOSUB7200:NEXT I
7040 RETURN
7050 :
7100 REM **** GIRAR (DR,AG) ****
7110 POKE REGDAT,(PEEK(REGDAT)AND 1)OR DR
7120 PL=PA*DS
7130 FOR I=1 TO PL:GOSUB7200:NEXT I
7140 RETURN
7150 :
7200 REM **** IMPULSO ****
7210 POKE REGDAT,PEEK(REGDAT)OR 8
7220 POKE REGDAT,PEEK(REGDAT)AND 247
7230 RETURN
7240 :
8000 REM **** SIGUIENTE FRANJA ****
8010 DR=BW:DS=30:GOSUB7000:REM MOVER
8020 DR=WA:AG=90:GOSUB7100:REM GIRAR
8030 DR=FW:DS=WD:GOSUB7000:REM MOVER
8040 DR=WA:AG=90:GOSUB7100:REM GIRAR
8050 RETURN
9000 REM **** XPLOT ****
9010 FOR I=0 TO WD
9020 X1=XS/4:Y1=(Y+I)/4:X2=X/4:Y2=Y1
9030 GOSUB 6000:REM ENTRAR LINESUB
9035 NEXT I
9040 RETURN
9100 REM **** YPLOT ****
9110 FOR I=0 TO WD
9120 X1=(X+I)/4:Y1=YS/4:X2=X1:Y2=Y/4
9130 GOSUB6000:REM ENTRAR LINESUB
9135 NEXT I
9140 RETURN
```

Punto de partida

Una vez estudiadas las llamadas OSBYTE, nos corresponde analizar las características de otro grupo de llamadas igualmente útiles: las OSWORD

Las llamadas OSBYTE tienen una limitación importante: sólo aceptan dos parámetros, uno en X y otro en Y. Contra lo cual nada hay que objetar siempre y cuando no se desee pasar un buen número de parámetros al sistema operativo, aunque en realidad, y aun poniéndole mucha voluntad, difícilmente se pueden controlar todas las rutinas con tan sólo dos parámetros. Por su parte, las rutinas OSWORD son capaces de controlar más de dos parámetros y resultan una utilísima alternativa a las llamadas OSBYTE.

OSWORD emplea un área de memoria conocida por *bloque de parámetros* para el paso de éstos al sistema operativo. Esta área, de tamaño variable según la llamada OSWORD que se está utilizando, se puede situar en cualquier lugar dentro de la RAM del usuario. Cuando se llama una rutina OSWORD, quien se encarga de decir cuál de las rutinas es la llamada es el registro A, mientras que los registros X e Y especifican la dirección del primer byte del bloque de parámetros. El registro X retiene el byte inferior de dicha dirección y el registro Y el superior. Por ejemplo, si el bloque se sitúa en la dirección &0A00, el registro X retiene 00 y el Y el valor &0A:

Ensamblador	BASIC
LDX #0 LDY #&0A	X%=0 Y%=&0A

Se produce la llamada OSWORD en la dirección &FFF1, que está vectorizada del mismo modo que una llamada OSBYTE. Pero el vector OSWORD se encuentra en la dirección &20C. Por tanto, para llamar una rutina OSWORD, especificada con el registro A conteniendo el valor 1, estableceremos el bloque con los parámetros necesarios, y después llamaremos la rutina por medio de las siguientes líneas en código (parameter_block es la dirección del primer byte del bloque):

```
LDX   # parameter_block MOD 256
LDY   # parameter_block DIV 256
LDA   #1
JSR   &FFF1
```

Naturalmente los valores del bloque de parámetros dependerán de la llamada concreta que se utilice.

¿Qué hacen las llamadas OSWORD?

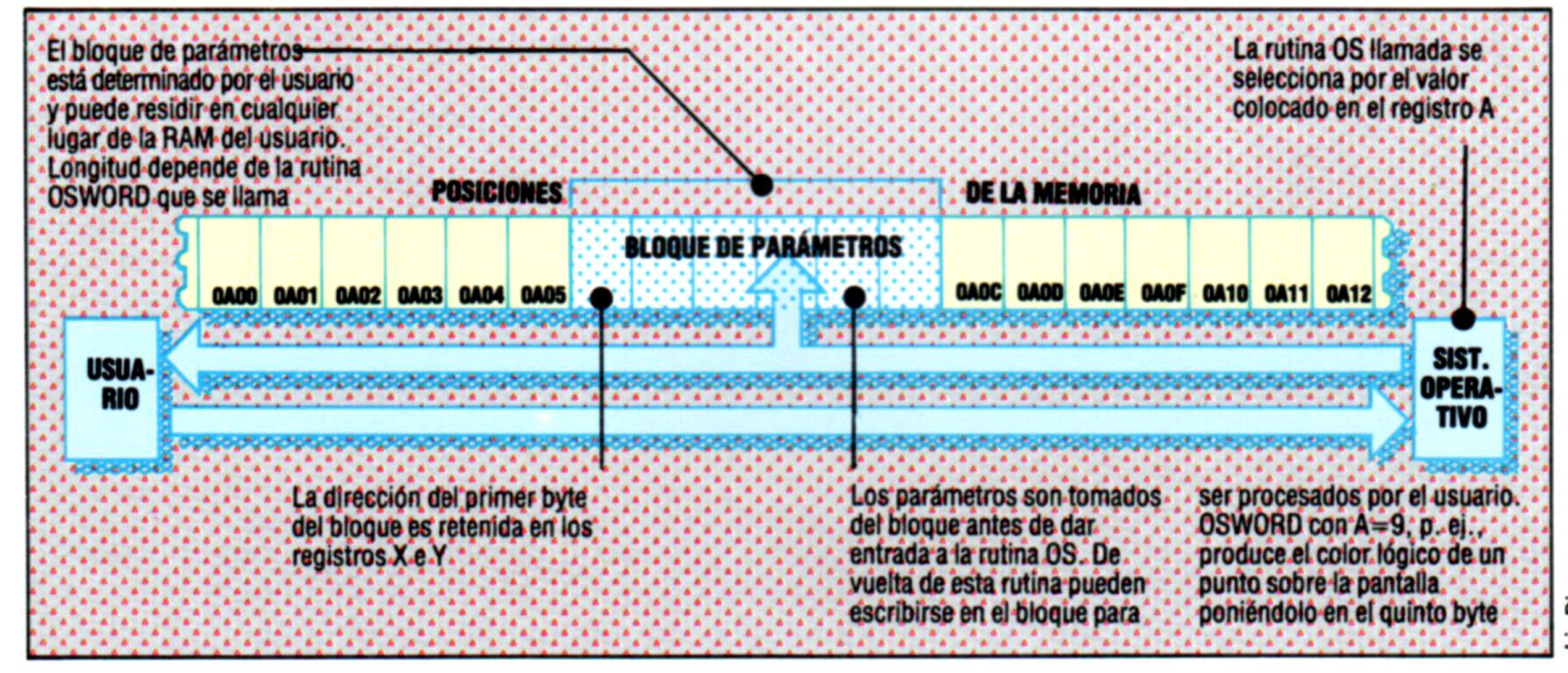
Con estas llamadas podemos leer información de la corriente actual de entrada y pasarla a la memoria, ejecutar instrucciones SOUND y ENVELOPE, acceder al sistema de ficheros en disco, y otras muchas funciones.

Veamos ahora cómo se utilizan algunas de estas llamadas OSWORD. Para identificarlas, se nombran según el contenido de A cuando se realiza la llamada a la dirección &FFF1: por ejemplo, OSWORD con A=0.

- *OSWORD con A=0*: Esta llamada nos permite leer una línea de entrada según la corriente seleccionada de entrada, lo que constituye una función de máxima utilidad. Nos permite especificar el número máximo de caracteres que serán aceptados y los valores máximo y mínimo permitidos en los códigos ASCII de los caracteres que han de entrar. Los datos leídos se almacenan en la memoria en la dirección especificada en los primeros dos bytes del bloque de parámetros.

Este bloque de la llamada OSWORD se establece de la siguiente manera:

Más pista
Una llamada OSWORD proporciona el medio más conveniente para controlar las rutinas del OS que necesitan más de los dos parámetros disponibles en las rutinas OSBYTE





Bloque de parámetros para OSWORD con A=0	
Byte	**Función**
0	Byte inferior de la dirección del buffer
1	Byte superior de la dirección del buffer
2	Número máximo de caracteres en la entrada
3	Máximo código ASCII aceptable
4	Mínimo código ASCII aceptable

Los registros X e Y señalan al byte 0 del bloque. Este programa escrito a continuación ilustra el uso de la llamada; el bloque se establece de tal manera que se puedan aceptar hasta 7 caracteres en código ASCII que vayan del 32 al 90, ambos inclusive.

```
 10 DIM C20
 20 A%=0
 30 X%=C MOD 256
 40 Y%=C DIV 256
 50 C?0=&00
 60 C?1=&0A:REM buffer en &0A00
 70 C?2=7
 80 C?3=32
 90 C?4=90
100 CALL &FFF1
110 PRINT $(&0A00)
```

Pulse RUN para ejecutarlo y digite algunos caracteres. La tecla Delete funcionará como siempre, y si pulsa Return o Escape provocará el final de esta rutina OSWORD. Es importante observar que si se intenta digitar caracteres cuyos códigos ASCII caen fuera del intervalo especificado, serán escritos en la pantalla pero no se agregarán a los caracteres escritos en el buffer. Si usted escribe demasiados caracteres, lo que oirá será un "pitido" de aviso. En lenguaje máquina, cuando se sale de la llamada OSWORD, el flag C se activa en caso de pulsar la tecla Escape; si esto no es así, C estará a 0. El registro Y retendrá el número de caracteres introducidos, incluyendo posibles retornos de carro.

• *OSWORD con A=1 y A=2*: Estas llamadas permiten al programador en código máquina acceder a la variable TIME del BASIC. La llamada OSWORD con A=1 nos permite leer el valor actual de TIME. Los cinco bytes que representan este valor se almacenan en el bloque de parámetros, que, obviamente, habrá de tener una longitud de cinco bytes para esta llamada. El byte menos significativo del valor es almacenado en el byte 0 del bloque y el más significativo en el byte 5.

OSWORD con A=2 nos permite dar a TIME un valor determinado. El byte menos significativo se coloca en el primer byte del bloque, el más significativo en el quinto, y seguidamente se realiza la llamada a OSWORD. El siguiente programa da a TIME el valor 100:

```
 10 DIM C 20
 20 X%=C MOD 256
 30 Y%=C DIV 256
 40 A%=2
 50 C?0=100
 60 C?1=0
 70 C?2=0
 80 C?3=0
 90 C?4=0
100 CALL &FFF1
```

• *OSWORD con A=3 y A=4*: El BBC posee un segundo reloj interno llamado *interval timer*, o bien *event timer*. No es empleado por TIME pero sí se utiliza por el *sistema de eventos* del ordenador, del que hablaremos más adelante. Este cronómetro comienza a contar, a la velocidad de un tic-tac cada centésima de segundo, partiendo de un determinado valor y genera un "evento" cuando llega a cero. Este evento se puede manipular fácilmente y puede servir para que el ordenador dé comienzo a una serie de operaciones transcurrido un tiempo determinado.

OSWORD con A=3 nos permite leer el valor actual del cronómetro *interval timer*, transfiriéndolo al bloque de parámetros, de un modo semejante a como OSWORD con A=1 leía el reloj de TIME. Para dar un determinado valor a este cronómetro se emplea OSWORD con A=4, así como con OSWORD con A=2 se daba un valor a TIME. Si, por ejemplo, hacemos que el bloque de parámetros retenga &FFFFFFFC, el evento ha de producirse transcurridos 0,004 segundos (4 milésimas de segundo).

• *OSWORD con A=5 y A=6*: Se trata de dos llamadas muy especiales, de extrema utilidad para aquellos programadores que tengan acceso al segundo procesador del BBC Micro. Una vez acoplado al ordenador uno de estos dispositivos, se comunicará a través de la interface Tube y considerará al ordenador como un procesador de E/S. Lo que significa que el BBC Micro actúa ahora como un dispositivo esclavo, realizando todas las tareas de servicio que le ordena el segundo procesador, como tratar las entradas del teclado, visualizar en pantalla, y toda función genérica de entrada/salida. Estas llamadas OSWORD permiten al segundo procesador el acceso a las posiciones de memoria del procesador de E/S.

Bloques de parámetros para OSWORD con A=5 y A=6		
Byte	**OSWORD con A=5**	**OSWORD con A=6**
0	BmnS de la dir. del proces. E/S	BmnS de la dir. del proces. E/S
1	Dirección del byte por leer	Dirección del byte por escribir
2	Dirección del byte por leer	Dirección del byte por escribir
3	BmsS del procesador E/S	BmsS del procesador E/S
4	Tras la llamada, es el byte contenido en la dirección superior	Byte que hay que escribir en la dirección superior

Este cuadro resume cómo se establece el bloque de parámetros para A=5 (de lectura de bytes de la memoria del procesador E/S) y para A=6 (de escritura).

Si la dirección sólo va a ser de 16 bits, que será el caso para el procesador de entrada/salida en un BBC Micro estándar, entonces la dirección se almacena con su byte inferior en (XY+0) y su byte superior en (XY+1).

• *OSWORD con A=7 y A=8*: Estas dos llamadas permiten al programador manipular el sonido con programas escritos tanto en lenguaje máquina como en BASIC. OSWORD con A=7 simula la orden SOUND del BASIC, y OSWORD con A=8 es el equivalente de una orden ENVELOPE. Veámoslas separadamente.

• *OSWORD con A=7*: Exige un bloque con ocho

bytes (véase su estructura en el cuadro) además de una indicación que le permita distinguir qué entrada del bloque corresponde a cada uno de los parámetros dados en la instrucción SOUND.

Bloque de parámetros para OSWORD con A=7	
Byte	**Función**
0	BmnS de NÚMERO CANAL
1	BmsS de NÚMERO CANAL
2	BmnS de AMPLITUD
3	BmsS de AMPLITUD
4	BmnS de TONO ("PITCH")
5	BmsS de TONO
6	BmnS de DURACIÓN
7	BmsS de DURACIÓN

El programa que ofrecemos a continuación ofrece la actuación de una llamada: simula la instrucción SOUND 1,−15,100,30.

```
 10 DIM C 20
 20 X%=C MOD 256
 30 Y%=C DIV 256
 40 A%=7
 50 C?0=1
 60 C?1=0:REM Canal 1
 70 C?2=-15 MOD 256
 80 C?3=-15 DIV 256
 90 C?4=100
100 C?5=0
110 C?6=30
115 C?7=0
120 CALL &FFF1
130 END
```

Naturalmente, este mismo efecto se consigue con muchísima mayor facilidad por medio de la misma instrucción SOUND, por lo que, en BASIC, esta llamada es de rarísimo uso.

• *OSWORD con A=8*: Esta llamada es todavía más complicada que OSWORD con A=7: ¡nada menos que 17 son los bytes que necesita en el bloque de parámetros!

Como siempre, los registros X e Y sirven para apuntar al bloque. La primera entrada dentro del bloque retiene el número de ENVELOPE, y los trece bytes restantes del bloque retienen los demás parámetros de ENVELOPE. Con toda seguridad usted encontrará mucho más fácil el empleo de la instrucción ENVELOPE desde BASIC para conseguir el sonido que desea.

Las llamadas OSWORD para gráficos

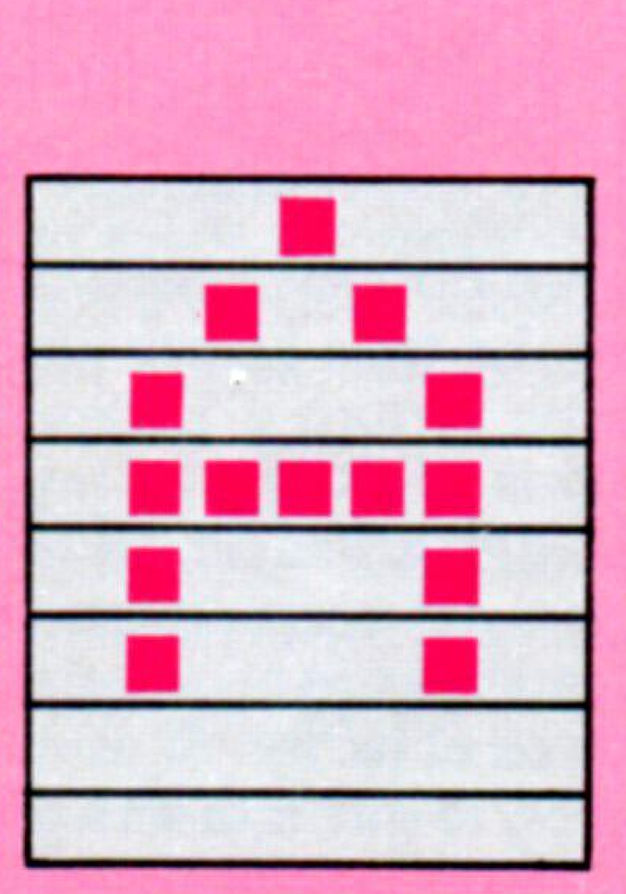
• *OSWORD con A=9*: Funciona de manera equivalente a la instrucción POINT del BASIC pero dentro de un programa en código máquina. Produce el color lógico en un punto dado, o el valor 255 si el punto especificado no está en pantalla. El valor se da en el quinto byte del bloque, que se establece así:

Bloque de parámetros para OSWORD con A=9	
Byte	**Función**
0	BmnS de la abscisa X
1	BmsS de la abscisa X
2	BmnS de la ordenada Y
3	BmsS de la ordenada Y
4	Color lógico

OSWORD con A=10

Si se desea imprimir caracteres grandes en pantalla, esta llamada es de mucha utilidad. Permite obtener la definición actual de cualquier carácter en Modos de 0 a 6. Es necesario un bloque de nueve bytes, como siempre apuntado por los registros X e Y. El primer byte del bloque deberá contener el código ASCII del carácter

Bloque de parámetros para A=10	
Byte	**Función**
0	Carácter escogido
1	Primera línea de defin.
2	Segunda línea
3	Tercera línea
4	Cuarta línea
5	Quinta línea
6	Sexta línea
7	Séptima línea
8	Última línea de defin.



Los valores producidos en el bloque de parámetros representan una expresión binaria de la definición del carácter donde 1 es un punto luminoso (pixel) sobre la pantalla y 0 es un pixel no iluminado

• *OSWORD con A=11*: Esta llamada nos permite inspeccionar los colores especificados en el conjunto de instrucciones de la VDU 19; nos permite encontrar el color *físico* asociado a un determinado color lógico. Lo cual puede resultar útil si se olvidaron los colores especificados en una determinada llamada a la VDU 19. Necesita un bloque de 5 bytes, y al realizarse la llamada el primer byte del bloque retendrá el valor del color lógico que nos interesa. Un ejemplo de su uso puede ser si suponemos ejecutada la orden VDU 19, 2, 4, 0, 0, 0. OSWORD con A=11 y un valor de 2 en el primer byte del bloque producirán los valores que mostramos al margen de esta página.

Byte	Valor
0	2
1	4
2	0
3	0
4	0

• *OSWORD con A=12*: Es una versión más rápida de VDU 19.

• *OSWORD con A=13*: Es otra llamada con aparentemente escasos usos. Produce las coordenadas X e Y de los últimos dos puntos repasados por el cursor de gráficos, lo que puede resultar práctico en ciertas ocasiones.

De este breve resumen usted puede colegir que las llamadas OSWORD son muy versátiles y proporcionan un fácil manejo de muchas facilidades del BBC Micro.



Con el fascículo anterior se han puesto a la venta las tapas correspondientes al sexto volumen

El juego de tapas va acompañado de un sobre con los transferibles, numerados del 1 al 8, correspondientes a los volúmenes de que consta la obra; esto le permitirá marcar el lomo de cada uno de los volúmenes, a medida que aumente su colección.

Para encuadernar los 12 fascículos que componen un volumen, es preciso arrancar previamente las cubiertas de los mismos.

No olvide que, antes de colocar los fascículos en las tapas intercambiables, debe usted estampar el número en el lomo de las mismas, siguiendo las instrucciones que se dan a continuación:



1 **Desprenda la hojita de protección y aplique el transferible en el lomo de la cubierta, haciendo coincidir los ángulos de referencia con los del recuadro del lomo.**

2 **Con un bolígrafo o un objeto de punta roma, repase varias veces el número, presionando como si quisiera borrarlo por completo.**

3 **Retire con cuidado y comprobará que el número ya está impreso en la cubierta. Cúbralo con la hojita de protección y repita la operación anterior con un objeto liso y redondeado, a fin de asegurar una perfecta y total adherencia.**

Editorial Delta

Cada sobre de transferibles contiene una serie completa de números, del 1 al 8, para fijar a los lomos de los volúmenes. Ya que en cada volumen sólo aplicará el número correspondiente, puede utilizar los restantes para hacer una prueba preliminar.

Ya están a su disposición, en todos los quioscos y librerías, las tapas intercambiables para encuadernar 12 fascículos de

mi COMPUTER

Cada juego de tapas va acompañado de una colección de transferibles, para que usted mismo pueda colocar en cada lomo el número de tomo que corresponda



9 788485 822836
10073